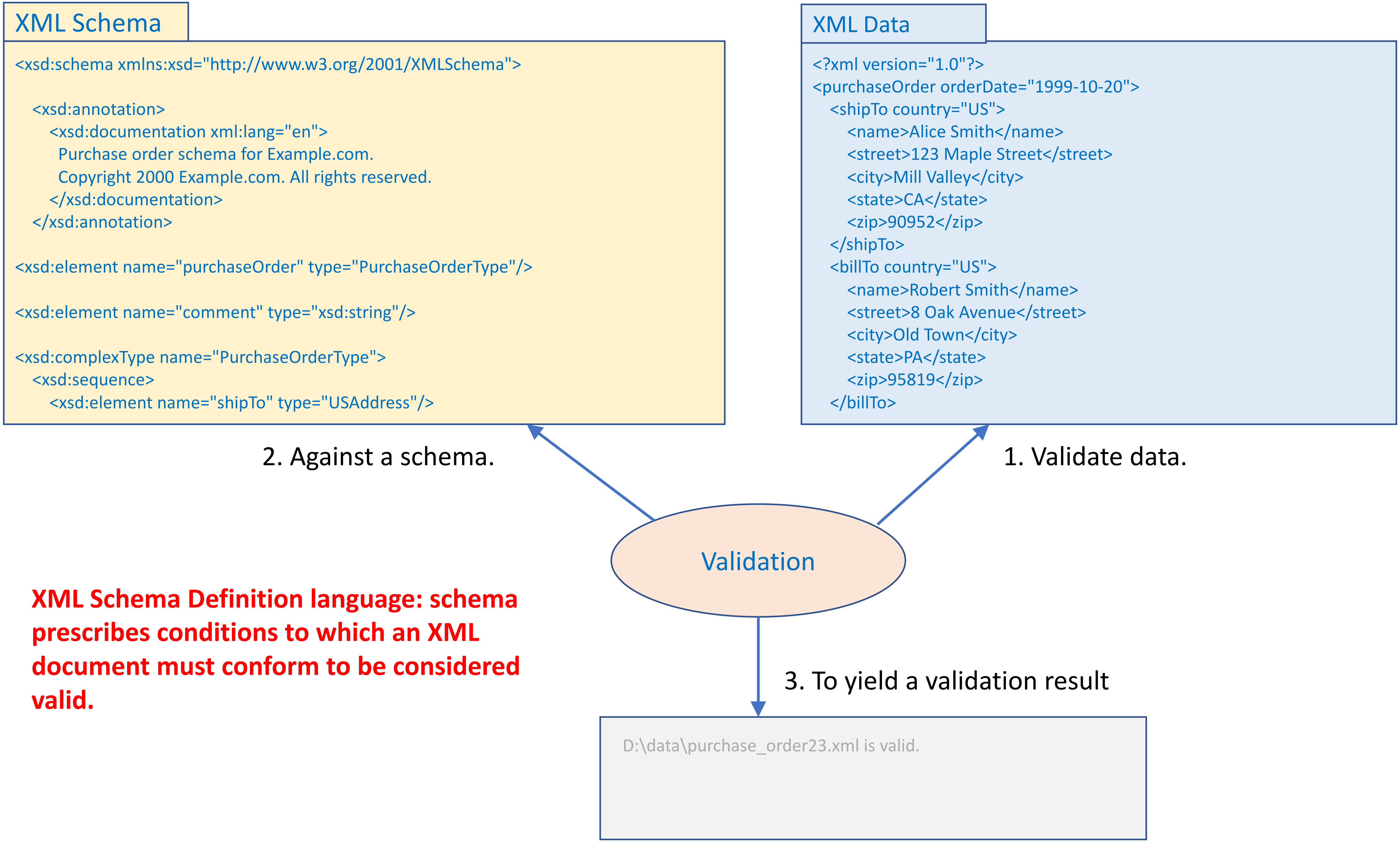
Testing data with the Shape Expressions language

Tom Baker

KIM Workshop 2018

Mannheim, 2018-04-10

https://shexspec.github.io/talks/blob/gh-pages/2018/04-10-kimworkshop-tombaker/ShEx.pdf



XML Schema

```
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema">

  <xsd:annotation>
    <xsd:documentation xml:lang="en">
     Purchase order schema for Example.com.
     Copyright 2000 Example.com. All rights reserved.
    </xsd:documentation>
  </xsd:annotation>

<xsd:element name="purchaseOrder" type="PurchaseOrderType"/>

<xsd:element name="comment" type="xsd:string"/>

<xsd:complexType name="PurchaseOrderType">
  <xsd:sequence>
    <xsd:element name="shipTo" type="USAddress"/>
```

XML Data

```
<?xml version="1.0"?>
<purchaseOrder orderDate="1999-10-20">
  <shipTo country="US">
    <name>Alice Smith</name>
    <street>123 Maple Street</street>
    <city>Mill Valley</city>
    <state>CA</state>
    <zip>90952</zip>
  </shipTo>
  <billTo country="US">
    <name>Robert Smith</name>
    <street>8 Oak Avenue</street>
    <city>Old Town</city>
    <state>PA</state>
    <zip>95819</zip>
  </billTo>
```

1. Validate data.

2. Against a schema.

Validation

3. To yield a validation result

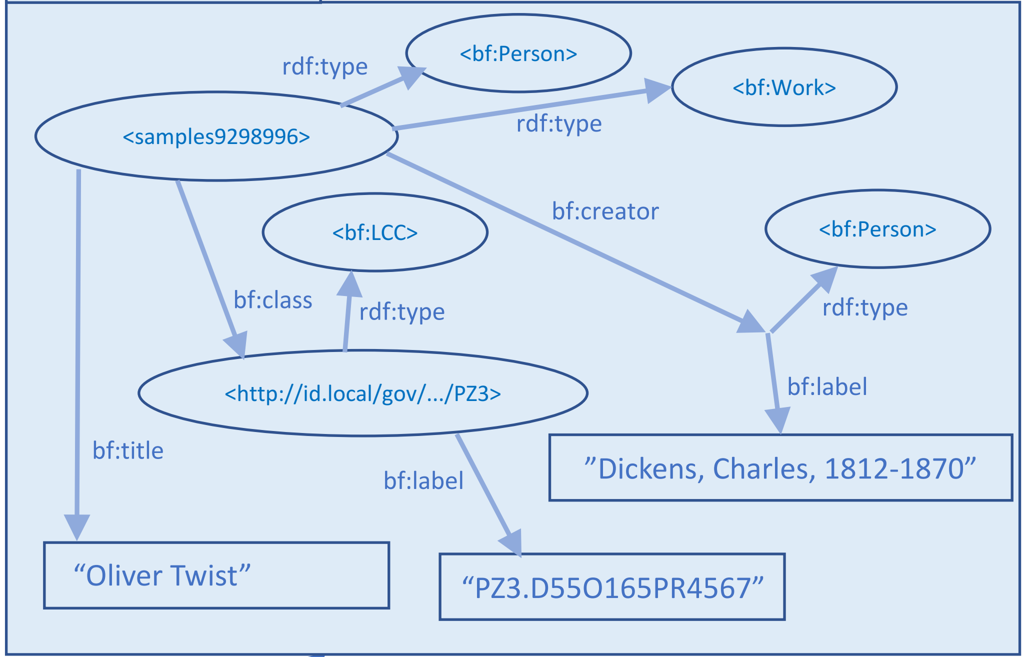
```
D:\data\purchase_order23.xml is valid.
```

XML Schema Definition language: schema prescribes conditions to which an XML document must conform to be considered valid.

ShEx Schema [ShExC syntax]

```
<Work> EXTRA rdf:type {
  rdf:type [bf:Work] ? ;
  bf:title LITERAL ;
  bf:class @<Classification> * ;
  bf:creator @<Person> OR @<Organization> + ;
  bf:derivedFrom IRI * ;
}

<Classification>
  [<http://id.loc.gov/…/>~]
AND
  EXTRA rdf:type {
    rdf:type [bf:LCC] ? ;
    bf:label LITERAL ;
}
```

RDF Graph



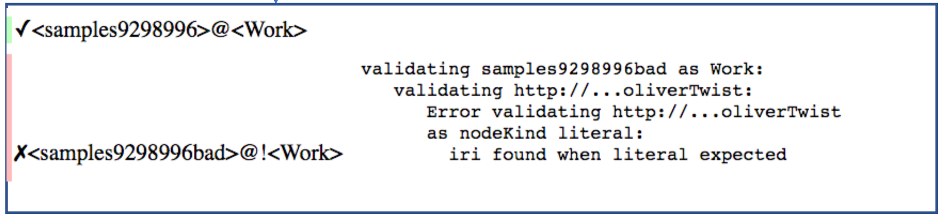
2. Against a schema.

1. Validate graph.

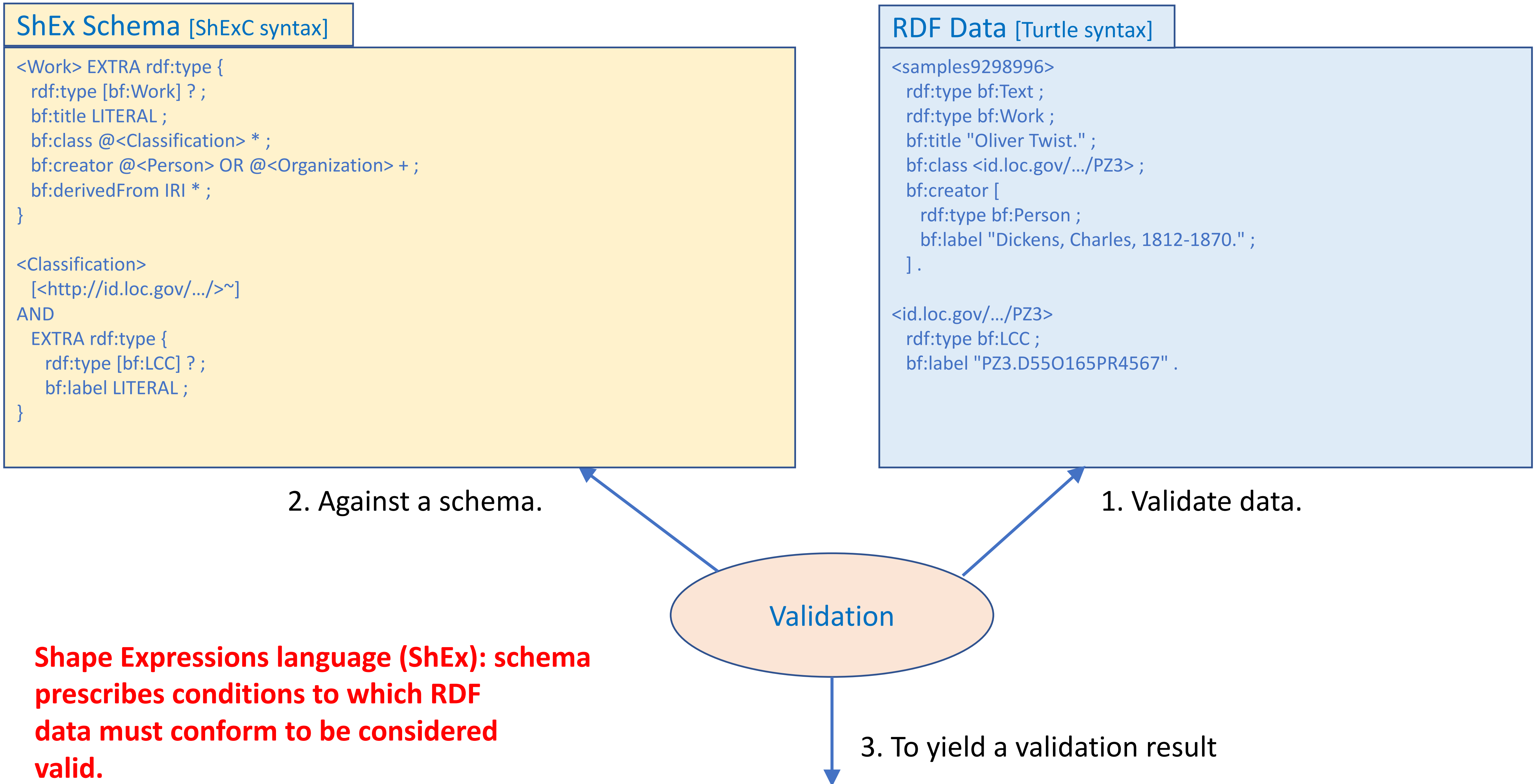
Validation

Shape Expressions language (ShEx): schema prescribes conditions to which an RDF graph must conform to be considered valid.

3. To yield a validation result



```
✓<samples9298996>@<Work>
                                    validating samples9298996bad as Work:
                                      validating http://...oliverTwist:
                                        Error validating http://...oliverTwist
                                        as nodeKind literal:
✗<samples9298996bad>@!<Work>              iri found when literal expected
```



ShEx Schema [ShExC syntax]

```
<Work> EXTRA rdf:type {
  rdf:type [bf:Work] ? ;
  bf:title LITERAL ;
  bf:class @<Classification> * ;
  bf:creator @<Person> OR @<Organization> + ;
  bf:derivedFrom IRI * ;
}

<Classification>
  [<http://id.loc.gov/…/>~]
AND
  EXTRA rdf:type {
    rdf:type [bf:LCC] ? ;
    bf:label LITERAL ;
}
```

RDF Data [Turtle syntax]

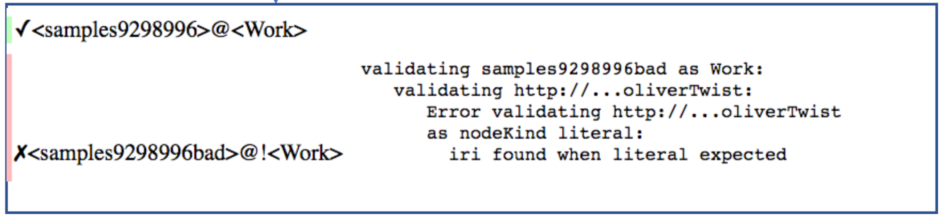
```
<samples9298996>
  rdf:type bf:Text ;
  rdf:type bf:Work ;
  bf:title "Oliver Twist." ;
  bf:class <id.loc.gov/…/PZ3> ;
  bf:creator [
    rdf:type bf:Person ;
    bf:label "Dickens, Charles, 1812-1870." ;
  ] .

<id.loc.gov/…/PZ3>
  rdf:type bf:LCC ;
  bf:label "PZ3.D55O165PR4567" .
```

1. Validate data.

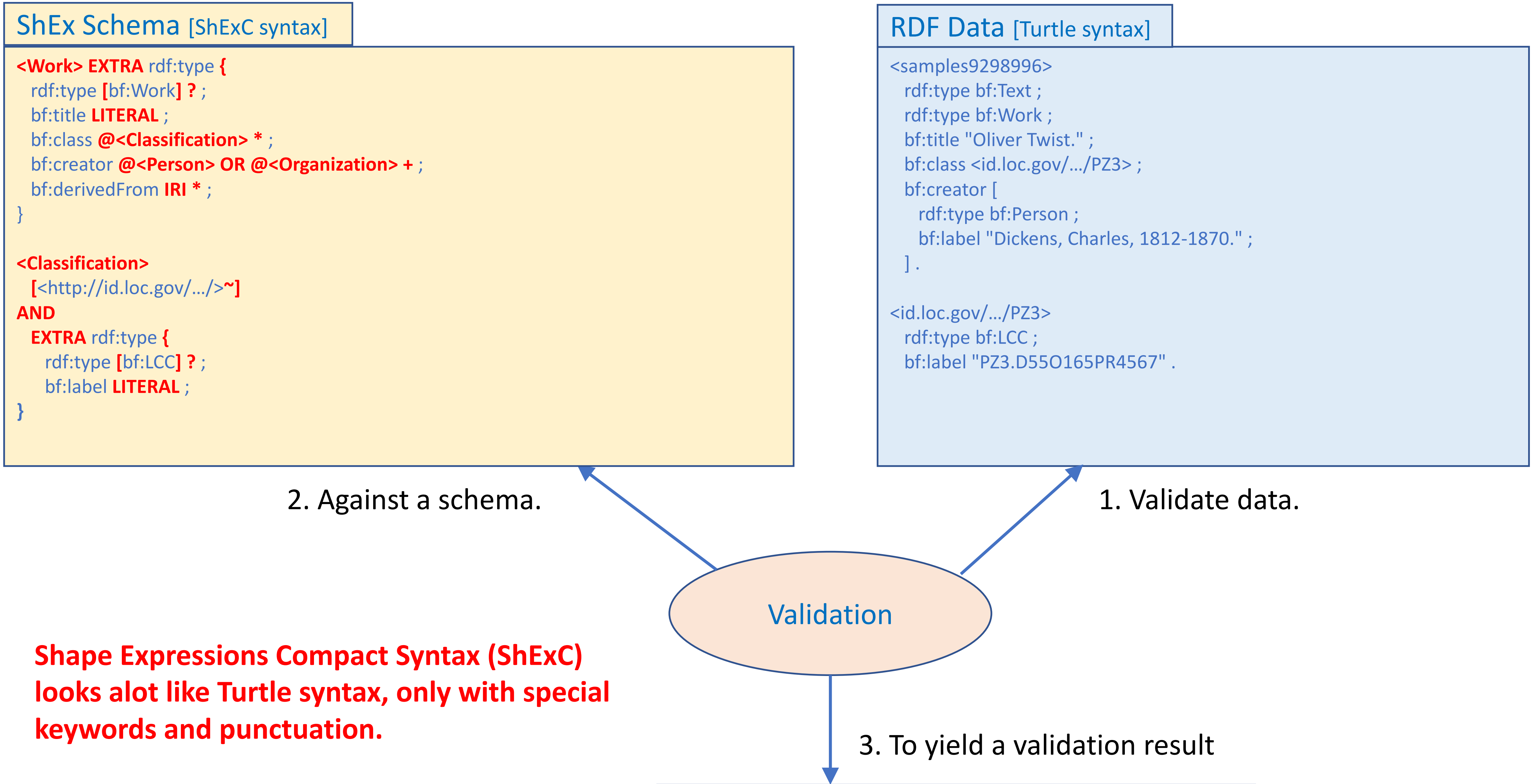
2. Against a schema.

3. To yield a validation result

Shape Expressions language (ShEx): schema prescribes conditions to which RDF data must conform to be considered valid.



```
✓<samples9298996>@<Work>
                              validating samples9298996bad as Work:
                                 validating http://...oliverTwist:
                                    Error validating http://...oliverTwist
                                    as nodeKind literal:
✗<samples9298996bad>@!<Work>          iri found when literal expected
```



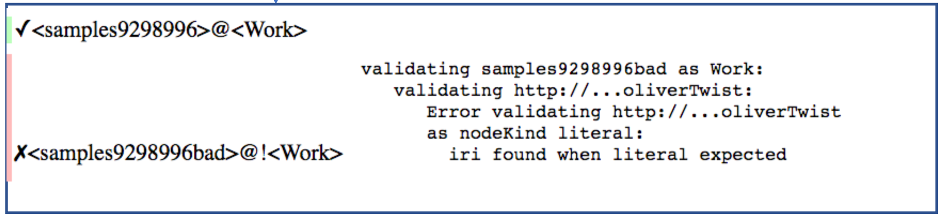
ShEx Schema [ShExC syntax]

```
<Work> EXTRA rdf:type {
  rdf:type [bf:Work] ? ;
  bf:title LITERAL ;
  bf:class @<Classification> * ;
  bf:creator @<Person> OR @<Organization> + ;
  bf:derivedFrom IRI * ;
}

<Classification>
  [<http://id.loc.gov/.../>~]
AND
  EXTRA rdf:type {
    rdf:type [bf:LCC] ? ;
    bf:label LITERAL ;
}
```

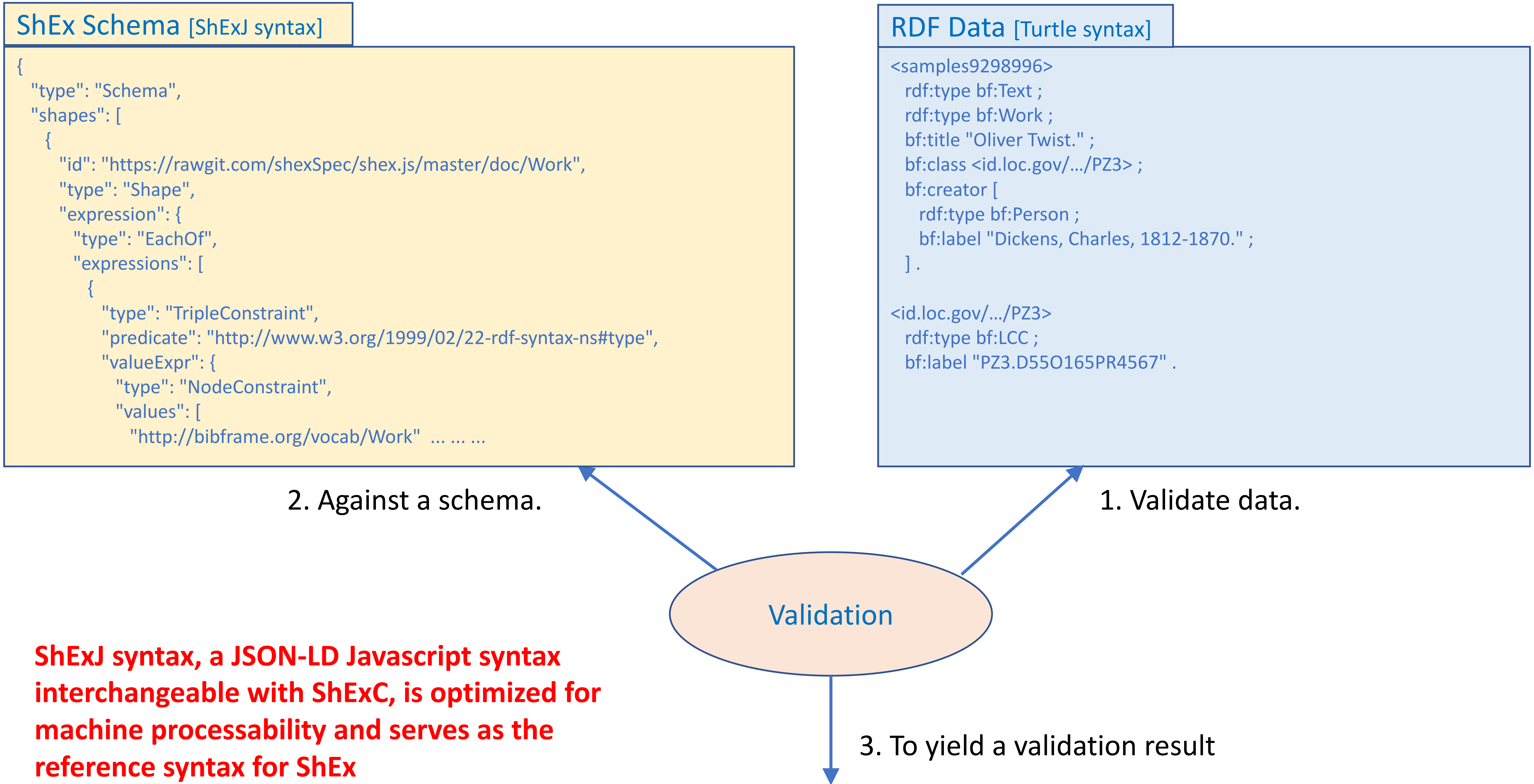
RDF Data [Turtle syntax]

```
<samples9298996>
  rdf:type bf:Text ;
  rdf:type bf:Work ;
  bf:title "Oliver Twist." ;
  bf:class <id.loc.gov/.../PZ3> ;
  bf:creator [
    rdf:type bf:Person ;
    bf:label "Dickens, Charles, 1812-1870." ;
  ] .

<id.loc.gov/.../PZ3>
  rdf:type bf:LCC ;
  bf:label "PZ3.D55O165PR4567" .
```

Shape Expressions Compact Syntax (ShExC) looks alot like Turtle syntax, only with special keywords and punctuation.



```
✓<samples9298996>@<Work>
                                   validating samples9298996bad as Work:
                                     validating http://...oliverTwist:
                                       Error validating http://...oliverTwist
                                       as nodeKind literal:
✗<samples9298996bad>@!<Work>              iri found when literal expected
```



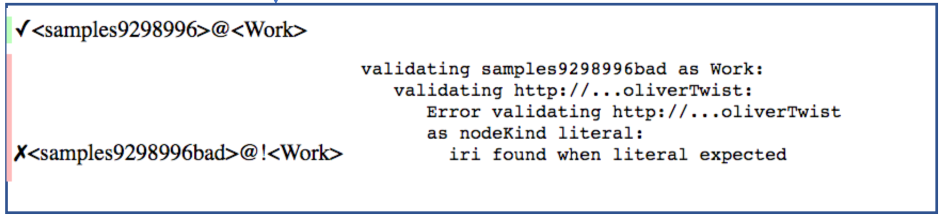
ShEx Schema [ShExJ syntax]

```
{
  "type": "Schema",
  "shapes": [
    {
      "id": "https://rawgit.com/shexSpec/shex.js/master/doc/Work",
      "type": "Shape",
      "expression": {
        "type": "EachOf",
        "expressions": [
          {
            "type": "TripleConstraint",
            "predicate": "http://www.w3.org/1999/02/22-rdf-syntax-ns#type",
            "valueExpr": {
              "type": "NodeConstraint",
              "values": [
                "http://bibframe.org/vocab/Work"  … … …
```

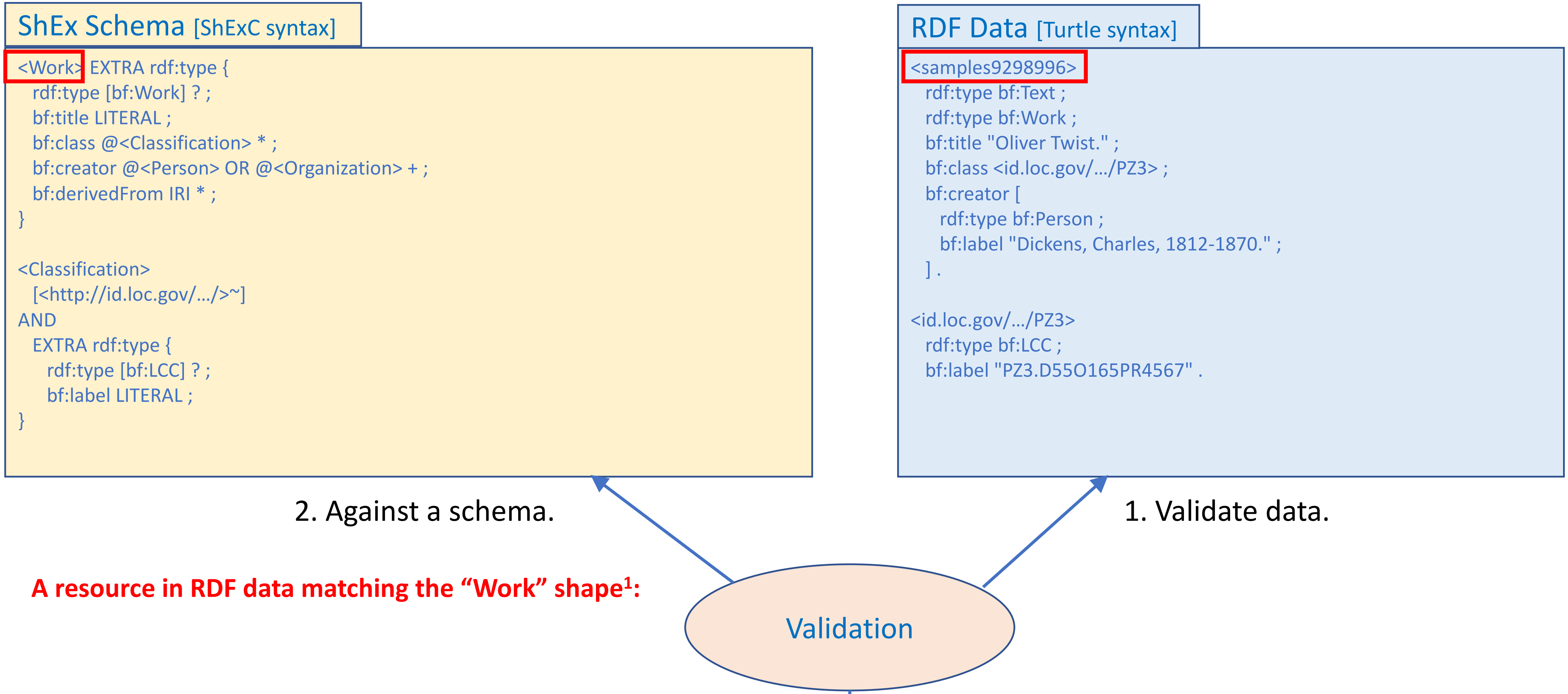
RDF Data [Turtle syntax]

```
<samples9298996>
  rdf:type bf:Text ;
  rdf:type bf:Work ;
  bf:title "Oliver Twist." ;
  bf:class <id.loc.gov/…/PZ3> ;
  bf:creator [
    rdf:type bf:Person ;
    bf:label "Dickens, Charles, 1812-1870." ;
  ] .

<id.loc.gov/…/PZ3>
  rdf:type bf:LCC ;
  bf:label "PZ3.D55O165PR4567" .
```

ShExJ syntax, a JSON-LD Javascript syntax interchangeable with ShExC, is optimized for machine processability and serves as the reference syntax for ShEx



```
✓<samples9298996>@<Work>
                                   validating samples9298996bad as Work:
                                     validating http://...oliverTwist:
                                       Error validating http://...oliverTwist
                                       as nodeKind literal:
✗<samples9298996bad>@!<Work>             iri found when literal expected
```



ShEx Schema [ShExC syntax]

```
<Work> EXTRA rdf:type {
  rdf:type [bf:Work] ? ;
  bf:title LITERAL ;
  bf:class @<Classification> * ;
  bf:creator @<Person> OR @<Organization> + ;
  bf:derivedFrom IRI * ;
}

<Classification>
  [<http://id.loc.gov/…/>~]
AND
  EXTRA rdf:type {
    rdf:type [bf:LCC] ? ;
    bf:label LITERAL ;
}
```

RDF Data [Turtle syntax]

```
<samples9298996>
  rdf:type bf:Text ;
  rdf:type bf:Work ;
  bf:title "Oliver Twist." ;
  bf:class <id.loc.gov/…/PZ3> ;
  bf:creator [
    rdf:type bf:Person ;
    bf:label "Dickens, Charles, 1812-1870." ;
  ] .

<id.loc.gov/…/PZ3>
  rdf:type bf:LCC ;
  bf:label "PZ3.D55O165PR4567" .
```

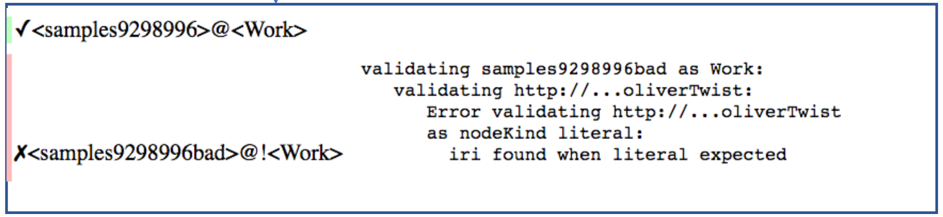
1. Validate data.

2. Against a schema.

Validation

A resource in RDF data matching the "Work" shape[1]:

3. To yield a validation result



```
✓<samples9298996>@<Work>

                              validating samples9298996bad as Work:
                                 validating http://...oliverTwist:
                                    Error validating http://...oliverTwist
                                    as nodeKind literal:
✗<samples9298996bad>@!<Work>           iri found when literal expected
```

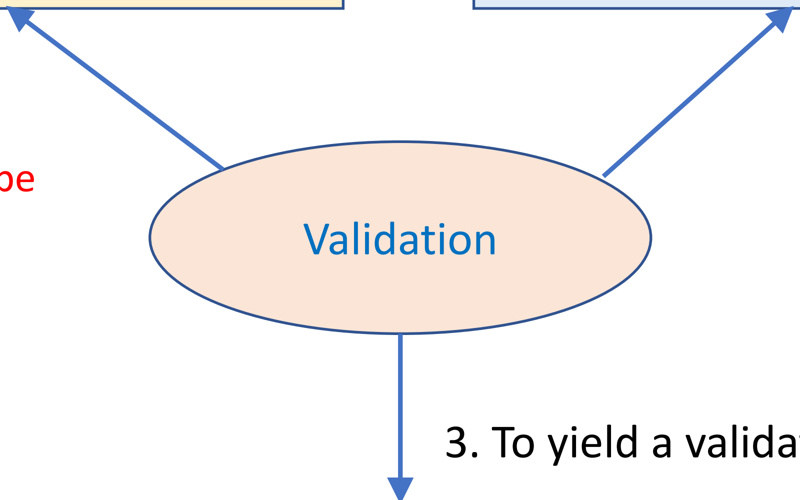
[1] Note: a pre-processing step uses the ShEx ShapeMap language to associate an RDF nodes with ShEx shapes, here: **`<samples9298996>@<Work>`**. Associations can be enumerated (in "fixed shape maps") or generated automatically (in "query shape maps") depending on the degree of control desired.

ShEx Schema [ShExC syntax]

```
<Work> EXTRA rdf:type {
  rdf:type [bf:Work] ? ;
  bf:title LITERAL ;
  bf:class @<Classification> * ;
  bf:creator @<Person> OR @<Organization> + ;
  bf:derivedFrom IRI * ;
}

<Classification>
  [<http://id.loc.gov/…/>~]
AND
  EXTRA rdf:type {
    rdf:type [bf:LCC] ? ;
    bf:label LITERAL ;
}
```

RDF Data [Turtle syntax]

```
<samples9298996>
  rdf:type bf:Text ;
  rdf:type bf:Work ;
  bf:title "Oliver Twist." ;
  bf:class <id.loc.gov/…/PZ3> ;
  bf:creator [
    rdf:type bf:Person ;
    bf:label "Dickens, Charles, 1812-1870." ;
  ] .

<id.loc.gov/…/PZ3>
  rdf:type bf:LCC ;
  bf:label "PZ3.D55O165PR4567" .
```



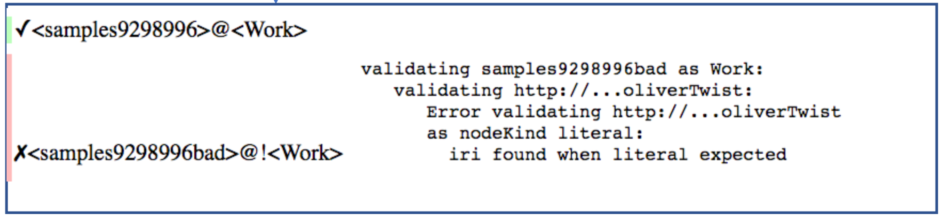
1. Validate data.

2. Against a schema.

3. To yield a validation result

A resource in RDF data matching the “Work” shape

Has zero or one “rdf:type bf:Work” statements



```
✓<samples9298996>@<Work>
                                        validating samples9298996bad as Work:
                                          validating http://...oliverTwist:
                                            Error validating http://...oliverTwist
                                            as nodeKind literal:
✗<samples9298996bad>@!<Work>                  iri found when literal expected
```

ShEx Schema [ShExC syntax]

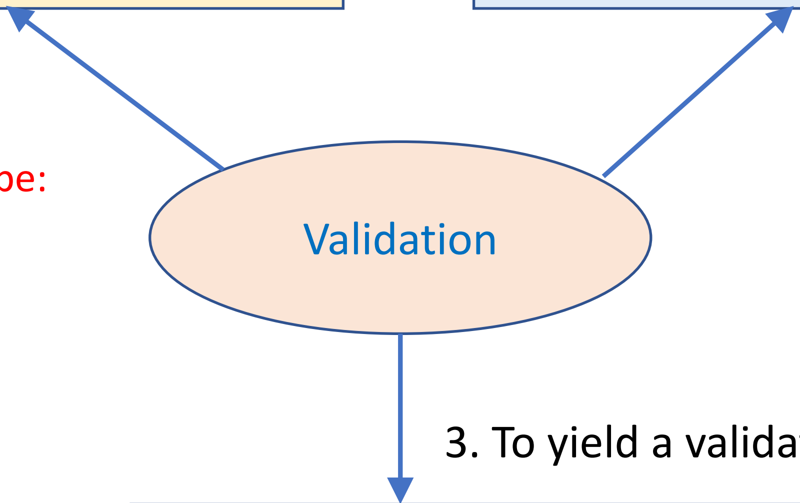
```
<Work> EXTRA rdf:type {
  rdf:type [bf:Work] ? ;
  bf:title LITERAL ;
  bf:class @<Classification> * ;
  bf:creator @<Person> OR @<Organization> + ;
  bf:derivedFrom IRI * ;
}

<Classification>
  [<http://id.loc.gov/…/>~]
AND
  EXTRA rdf:type {
    rdf:type [bf:LCC] ? ;
    bf:label LITERAL ;
}
```

RDF Data [Turtle syntax]

```
<samples9298996>
  rdf:type bf:Text ;
  rdf:type bf:Work ;
  bf:title "Oliver Twist." ;
  bf:class <id.loc.gov/…/PZ3> ;
  bf:creator [
    rdf:type bf:Person ;
    bf:label "Dickens, Charles, 1812-1870." ;
  ] .

<id.loc.gov/…/PZ3>
  rdf:type bf:LCC ;
  bf:label "PZ3.D55O165PR4567" .
```

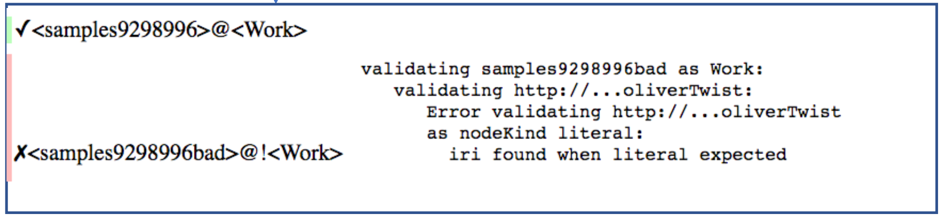
2. Against a schema.

1. Validate data.

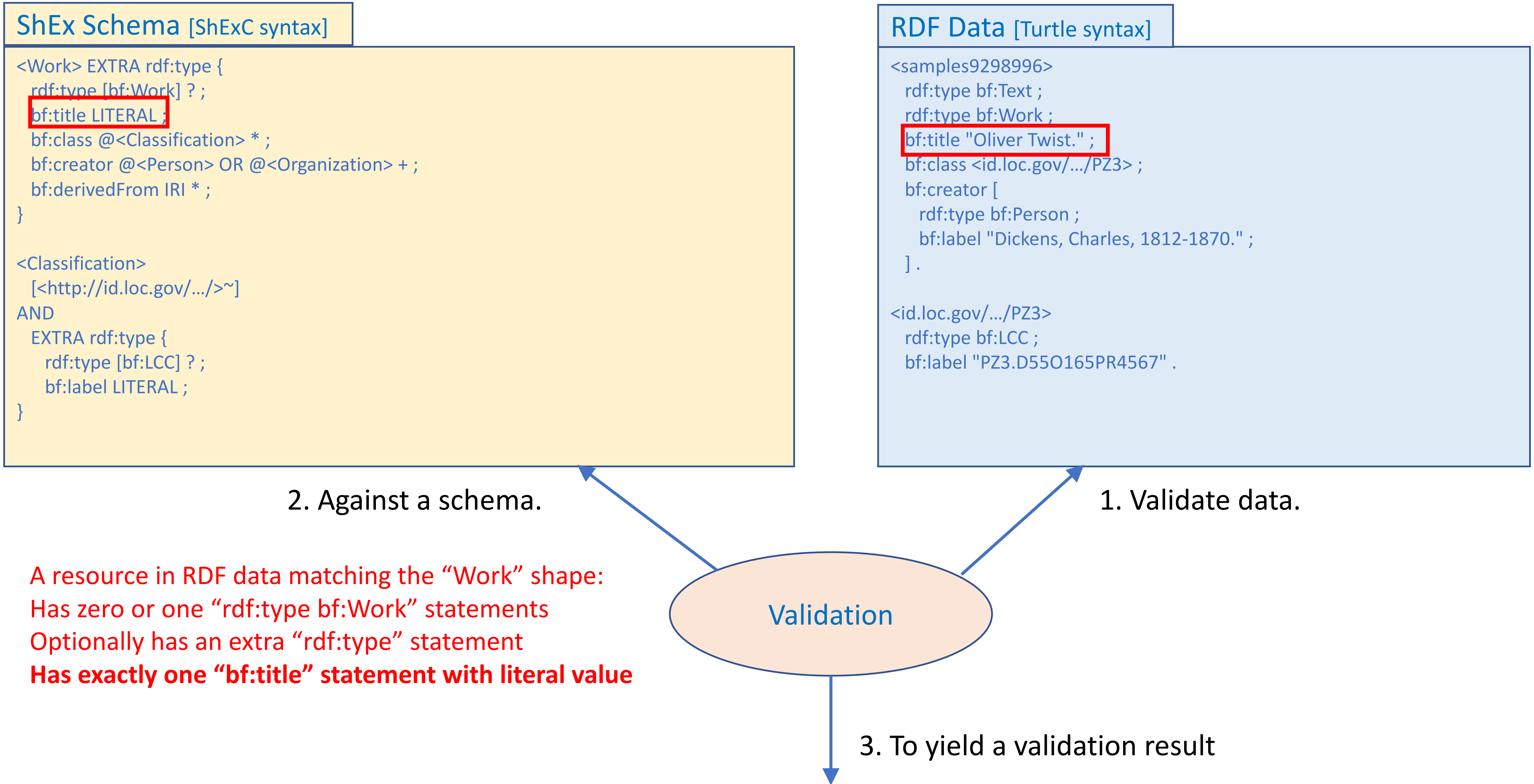


A resource in RDF data matching the “Work” shape:
Has zero or one “rdf:type bf:Work” statements
Optionally has an extra “rdf:type” statement

3. To yield a validation result



```
✓<samples9298996>@<Work>

                         validating samples9298996bad as Work:
                            validating http://...oliverTwist:
                               Error validating http://...oliverTwist
                               as nodeKind literal:
✗<samples9298996bad>@!<Work>       iri found when literal expected
```



ShEx Schema [ShExC syntax]

```
<Work> EXTRA rdf:type {
  rdf:type [bf:Work] ? ;
  bf:title LITERAL ;
  bf:class @<Classification> * ;
  bf:creator @<Person> OR @<Organization> + ;
  bf:derivedFrom IRI * ;
}

<Classification>
  [<http://id.loc.gov/…/>~]
AND
  EXTRA rdf:type {
    rdf:type [bf:LCC] ? ;
    bf:label LITERAL ;
}
```

RDF Data [Turtle syntax]

```
<samples9298996>
  rdf:type bf:Text ;
  rdf:type bf:Work ;
  bf:title "Oliver Twist." ;
  bf:class <id.loc.gov/…/PZ3> ;
  bf:creator [
    rdf:type bf:Person ;
    bf:label "Dickens, Charles, 1812-1870." ;
  ] .

<id.loc.gov/…/PZ3>
  rdf:type bf:LCC ;
  bf:label "PZ3.D55O165PR4567" .
```

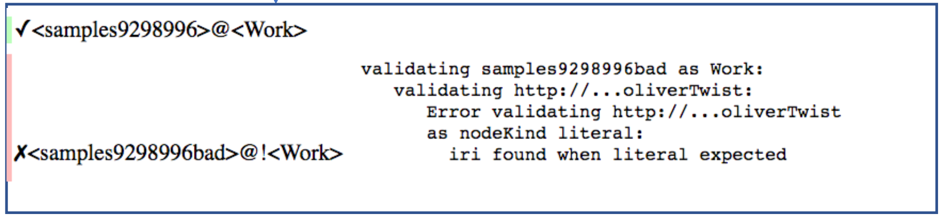
2. Against a schema.

1. Validate data.

Validation

3. To yield a validation result

A resource in RDF data matching the “Work” shape:
Has zero or one “rdf:type bf:Work” statements
Optionally has an extra “rdf:type” statement
Has exactly one “bf:title” statement with literal value



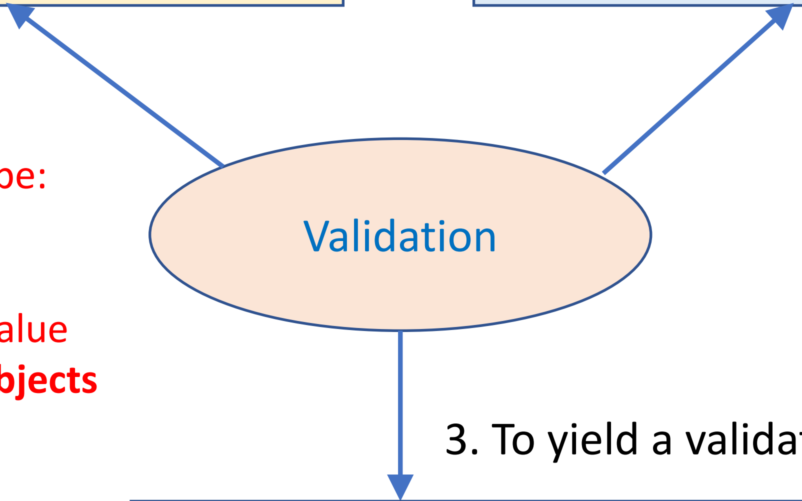
```
✓<samples9298996>@<Work>
                                   validating samples9298996bad as Work:
                                     validating http://...oliverTwist:
                                       Error validating http://...oliverTwist
                                       as nodeKind literal:
✗<samples9298996bad>@!<Work>             iri found when literal expected
```

ShEx Schema [ShExC syntax]

```
<Work> EXTRA rdf:type {
  rdf:type [bf:Work] ? ;
  bf:title LITERAL ;
  bf:class @<Classification> * ;
  bf:creator @<Person> OR @<Organization> + ;
  bf:derivedFrom IRI * ;
}

<Classification>
  [<http://id.loc.gov/…/>~]
AND
  EXTRA rdf:type {
    rdf:type [bf:LCC] ? ;
    bf:label LITERAL ;
}
```

RDF Data [Turtle syntax]

```
<samples9298996>
  rdf:type bf:Text ;
  rdf:type bf:Work ;
  bf:title "Oliver Twist." ;
  bf:class <id.loc.gov/…/PZ3> ;
  bf:creator [
    rdf:type bf:Person ;
    bf:label "Dickens, Charles, 1812-1870." ;
  ] .

<id.loc.gov/…/PZ3>
  rdf:type bf:LCC ;
  bf:label "PZ3.D55O165PR4567" .
```



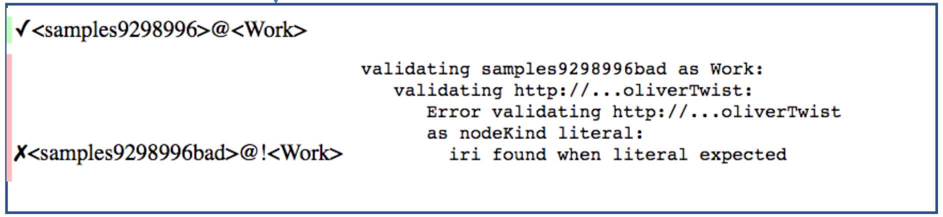
2. Against a schema.

1. Validate data.

Validation

3. To yield a validation result

A resource in RDF data matching the “Work” shape:
Has zero or one “rdf:type bf:Work” statements
Optionally has an extra “rdf:type” statement
Has exactly one “bf:title” statement with literal value
Has zero or more “bf:class” statements taking objects that match the “Classification” shape



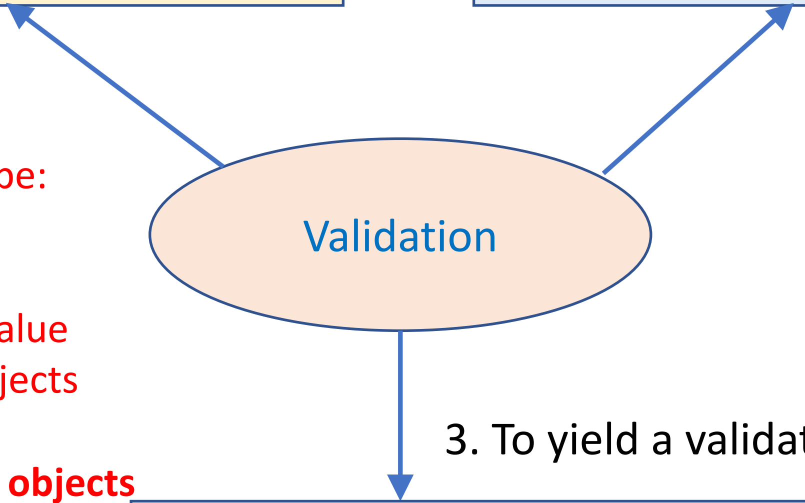
```
✓<samples9298996>@<Work>
                                   validating samples9298996bad as Work:
                                     validating http://...oliverTwist:
                                       Error validating http://...oliverTwist
                                       as nodeKind literal:
✗<samples9298996bad>@!<Work>             iri found when literal expected
```

ShEx Schema [ShExC syntax]

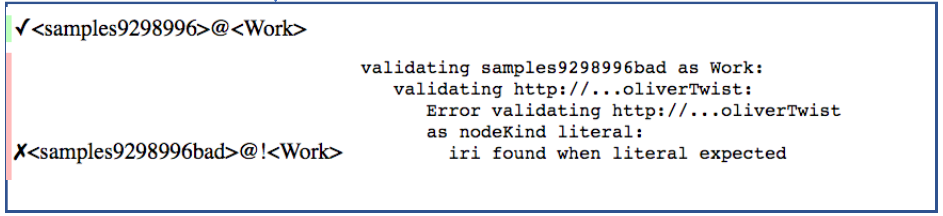
```
<Work> EXTRA rdf:type {
  rdf:type [bf:Work] ? ;
  bf:title LITERAL ;
  bf:class @<Classification> * ;
  bf:creator @<Person> OR @<Organization> + ;
  bf:derivedFrom IRI * ;
}

<Classification>
  [<http://id.loc.gov/…/>~]
AND
  EXTRA rdf:type {
    rdf:type [bf:LCC] ? ;
    bf:label LITERAL ;
}
```

RDF Data [Turtle syntax]

```
<samples9298996>
  rdf:type bf:Text ;
  rdf:type bf:Work ;
  bf:title "Oliver Twist." ;
  bf:class <id.loc.gov/…/PZ3> ;
  bf:creator [
    rdf:type bf:Person ;
    bf:label "Dickens, Charles, 1812-1870." ;
  ] .

<id.loc.gov/…/PZ3>
  rdf:type bf:LCC ;
  bf:label "PZ3.D55O165PR4567" .
```



A resource in RDF data matching the “Work” shape:
Has zero or one “rdf:type bf:Work” statements
Optionally has an extra “rdf:type” statement
Has exactly one “bf:title” statement with literal value
Has zero or more “bf:class” statements taking objects that match the “Classification” shape
Has one or more “bf:creator” statements taking objects that match either the “Person” or the “Organization” shape



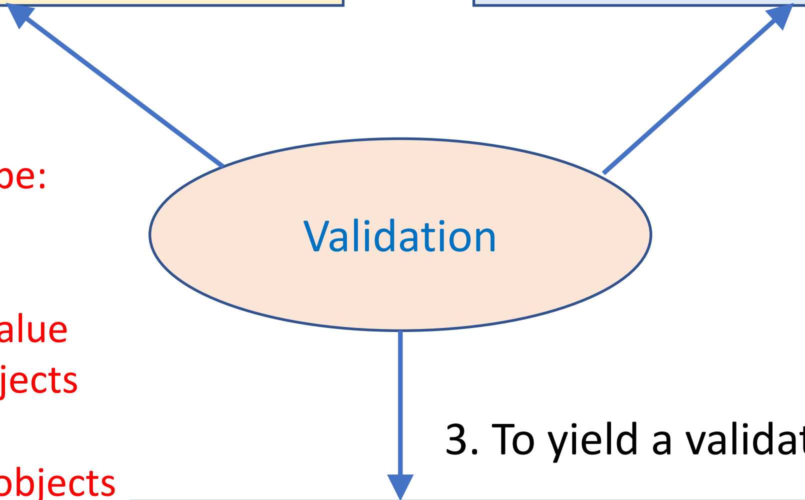
```
✓<samples9298996>@<Work>
                                   validating samples9298996bad as Work:
                                     validating http://...oliverTwist:
                                       Error validating http://...oliverTwist
                                       as nodeKind literal:
✗<samples9298996bad>@!<Work>             iri found when literal expected
```

ShEx Schema [ShExC syntax]

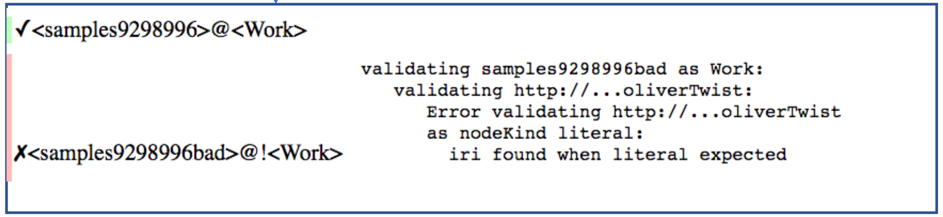
```
<Work> EXTRA rdf:type {
  rdf:type [bf:Work] ? ;
  bf:title LITERAL ;
  bf:class @<Classification> * ;
  bf:creator @<Person> OR @<Organization> + ;
  bf:derivedFrom IRI * ;
}

<Classification>
  [<http://id.loc.gov/…/>~]
AND
  EXTRA rdf:type {
    rdf:type [bf:LCC] ? ;
    bf:label LITERAL ;
}
```

RDF Data [Turtle syntax]

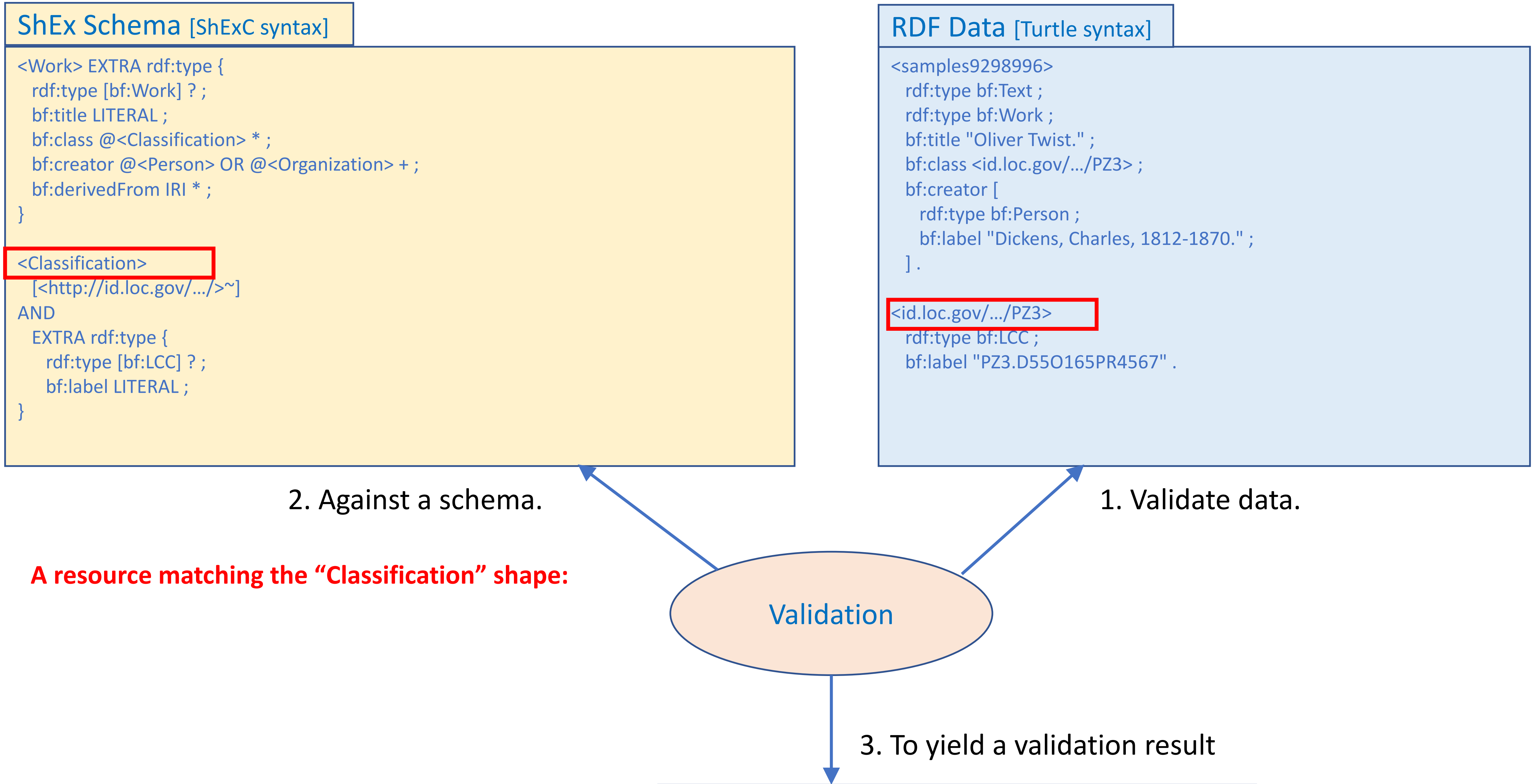
```
<samples9298996>
  rdf:type bf:Text ;
  rdf:type bf:Work ;
  bf:title "Oliver Twist." ;
  bf:class <id.loc.gov/…/PZ3> ;
  bf:creator [
    rdf:type bf:Person ;
    bf:label "Dickens, Charles, 1812-1870." ;
  ] .

<id.loc.gov/…/PZ3>
  rdf:type bf:LCC ;
  bf:label "PZ3.D55O165PR4567" .
```



A resource in RDF data matching the “Work” shape:
Has zero or one “rdf:type bf:Work” statements
Optionally has an extra “rdf:type” statement
Has exactly one “bf:title” statement with literal value
Has zero or more “bf:class” statements taking objects that match the “Classification” shape
Has one or more “bf:creator” statements taking objects that match either the “Person” or the “Organization” Shape
And zero or more “bf:derivedFrom” statements taking an IRI as object.



```
✓<samples9298996>@<Work>
                                  validating samples9298996bad as Work:
                                    validating http://...oliverTwist:
                                      Error validating http://...oliverTwist
                                      as nodeKind literal:
✗<samples9298996bad>@!<Work>            iri found when literal expected
```



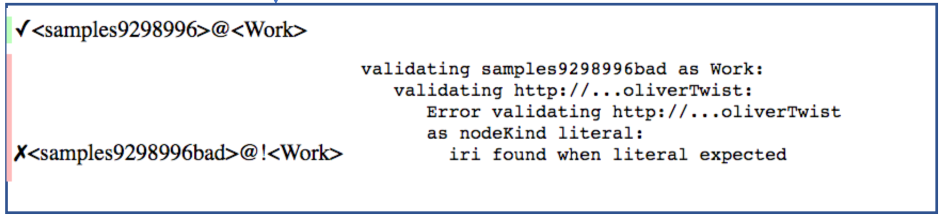
ShEx Schema [ShExC syntax]

```
<Work> EXTRA rdf:type {
  rdf:type [bf:Work] ? ;
  bf:title LITERAL ;
  bf:class @<Classification> * ;
  bf:creator @<Person> OR @<Organization> + ;
  bf:derivedFrom IRI * ;
}

<Classification>
  [<http://id.loc.gov/.../>~]
AND
  EXTRA rdf:type {
    rdf:type [bf:LCC] ? ;
    bf:label LITERAL ;
}
```

RDF Data [Turtle syntax]

```
<samples9298996>
  rdf:type bf:Text ;
  rdf:type bf:Work ;
  bf:title "Oliver Twist." ;
  bf:class <id.loc.gov/.../PZ3> ;
  bf:creator [
    rdf:type bf:Person ;
    bf:label "Dickens, Charles, 1812-1870." ;
  ] .

<id.loc.gov/.../PZ3>
  rdf:type bf:LCC ;
  bf:label "PZ3.D55O165PR4567" .
```

A resource matching the “Classification” shape:



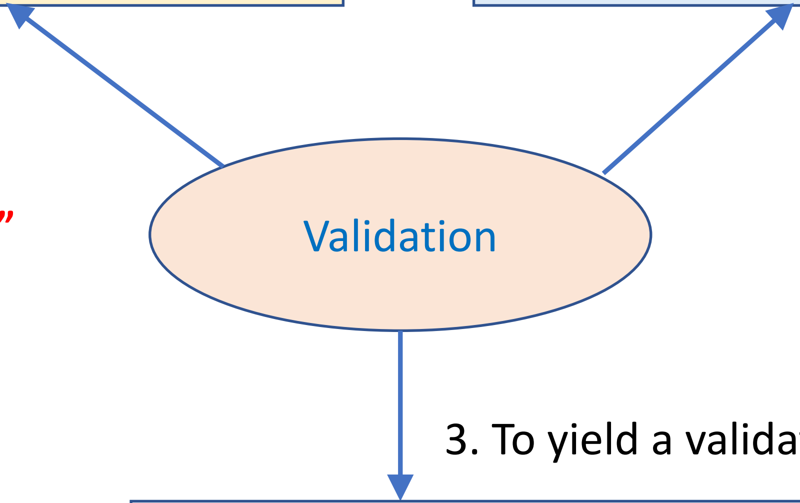
```
✓<samples9298996>@<Work>
                                  validating samples9298996bad as Work:
                                    validating http://...oliverTwist:
                                      Error validating http://...oliverTwist
                                      as nodeKind literal:
✗<samples9298996bad>@!<Work>            iri found when literal expected
```

ShEx Schema [ShExC syntax]

```
<Work> EXTRA rdf:type {
  rdf:type [bf:Work] ? ;
  bf:title LITERAL ;
  bf:class @<Classification> * ;
  bf:creator @<Person> OR @<Organization> + ;
  bf:derivedFrom IRI * ;
}

<Classification>
  [<http://id.loc.gov/.../>~]
AND
  EXTRA rdf:type {
    rdf:type [bf:LCC] ? ;
    bf:label LITERAL ;
}
```

RDF Data [Turtle syntax]

```
<samples9298996>
  rdf:type bf:Text ;
  rdf:type bf:Work ;
  bf:title "Oliver Twist." ;
  bf:class <id.loc.gov/.../PZ3> ;
  bf:creator [
    rdf:type bf:Person ;
    bf:label "Dickens, Charles, 1812-1870." ;
  ] .

<id.loc.gov/.../PZ3>
  rdf:type bf:LCC ;
  bf:label "PZ3.D55O165PR4567" .
```

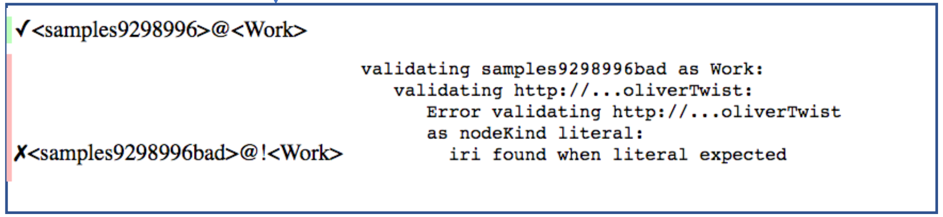


2. Against a schema.

1. Validate data.

A resource matching the “Classification” shape:
is identified with an IRI starting with “id.loc.gov”

3. To yield a validation result



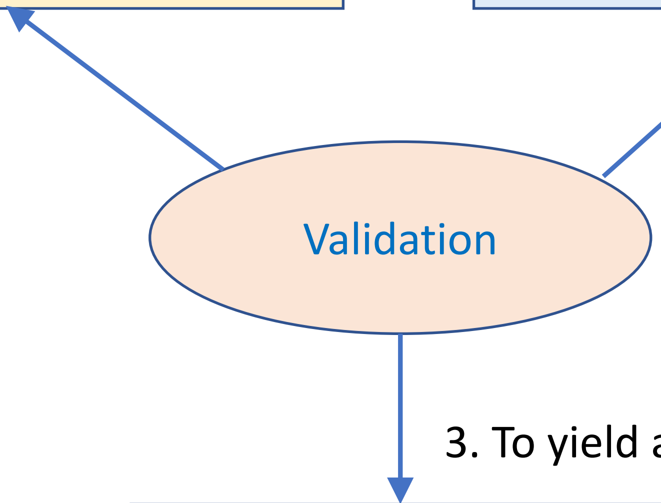
```
✓<samples9298996>@<Work>
                                    validating samples9298996bad as Work:
                                      validating http://...oliverTwist:
                                        Error validating http://...oliverTwist
                                        as nodeKind literal:
✗<samples9298996bad>@!<Work>              iri found when literal expected
```

ShEx Schema [ShExC syntax]

```
<Work> EXTRA rdf:type {
  rdf:type [bf:Work] ? ;
  bf:title LITERAL ;
  bf:class @<Classification> * ;
  bf:creator @<Person> OR @<Organization> + ;
  bf:derivedFrom IRI * ;
}

<Classification>
  [<http://id.loc.gov/…/>~]
AND
  EXTRA rdf:type {
    rdf:type [bf:LCC] ? ;
    bf:label LITERAL ;
}
```

RDF Data [Turtle syntax]

```
<samples9298996>
  rdf:type bf:Text ;
  rdf:type bf:Work ;
  bf:title "Oliver Twist." ;
  bf:class <id.loc.gov/…/PZ3> ;
  bf:creator [
    rdf:type bf:Person ;
    bf:label "Dickens, Charles, 1812-1870." ;
  ] .

<id.loc.gov/…/PZ3>
  rdf:type bf:LCC ;
  bf:label "PZ3.D55O165PR4567" .
```



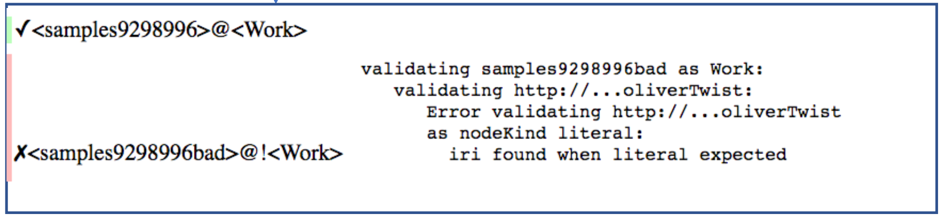
2. Against a schema.

1. Validate data.

Validation

3. To yield a validation result

A resource matching the “Classification” shape:
is identified with an IRI starting with “id.loc.gov”
has zero or one “rdf:type bf:LOC statements



✓<samples9298996>@<Work>

✗<samples9298996bad>@!<Work>

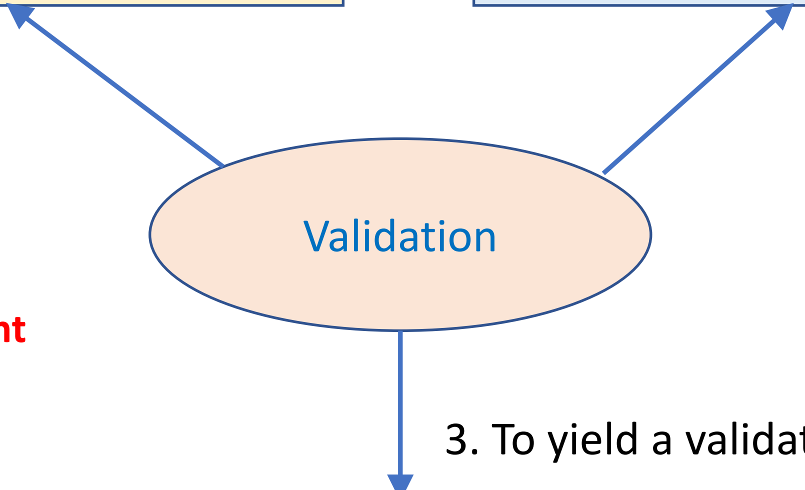
```
validating samples9298996bad as Work:
   validating http://...oliverTwist:
      Error validating http://...oliverTwist
      as nodeKind literal:
        iri found when literal expected
```

ShEx Schema [ShExC syntax]

```
<Work> EXTRA rdf:type {
  rdf:type [bf:Work] ? ;
  bf:title LITERAL ;
  bf:class @<Classification> * ;
  bf:creator @<Person> OR @<Organization> + ;
  bf:derivedFrom IRI * ;
}

<Classification>
  [<http://id.loc.gov/.../>~]
AND
  EXTRA rdf:type {
    rdf:type [bf:LCC] ? ;
    bf:label LITERAL ;
}
```

RDF Data [Turtle syntax]

```
<samples9298996>
  rdf:type bf:Text ;
  rdf:type bf:Work ;
  bf:title "Oliver Twist." ;
  bf:class <id.loc.gov/.../PZ3> ;
  bf:creator [
    rdf:type bf:Person ;
    bf:label "Dickens, Charles, 1812-1870." ;
  ] .

<id.loc.gov/.../PZ3>
  rdf:type bf:LCC ;
  bf:label "PZ3.D55O165PR4567" .
```

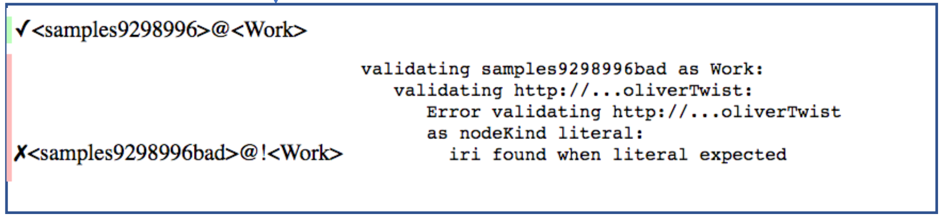


2. Against a schema.

1. Validate data.

A resource matching the “Classification” shape:
is identified with an IRI starting with “id.loc.gov”
has zero or one “rdf:type bf:LOC statements
optionally has an additional “rdf:type” statement

3. To yield a validation result



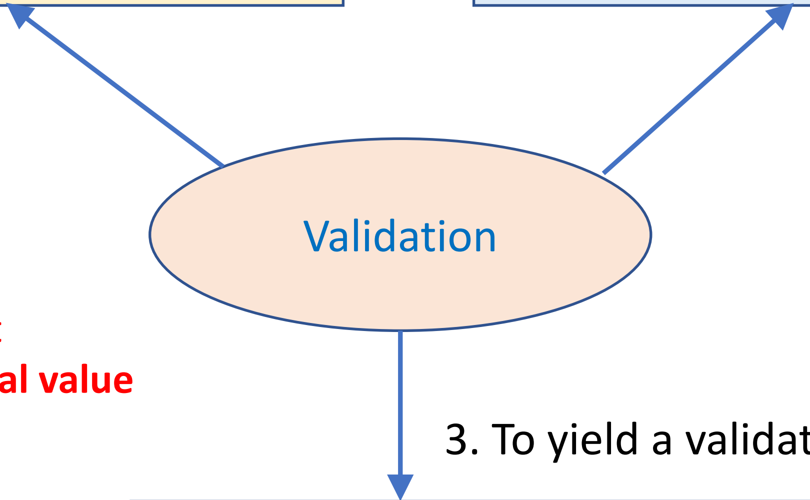
```
✓<samples9298996>@<Work>
                                      validating samples9298996bad as Work:
                                        validating http://...oliverTwist:
                                          Error validating http://...oliverTwist
                                          as nodeKind literal:
✗<samples9298996bad>@!<Work>                iri found when literal expected
```

ShEx Schema [ShExC syntax]

```
<Work> EXTRA rdf:type {
  rdf:type [bf:Work] ? ;
  bf:title LITERAL ;
  bf:class @<Classification> * ;
  bf:creator @<Person> OR @<Organization> + ;
  bf:derivedFrom IRI * ;
}

<Classification>
  [<http://id.loc.gov/…/>~]
AND
  EXTRA rdf:type {
    rdf:type [bf:LCC] ? ;
    bf:label LITERAL ;
}
```

RDF Data [Turtle syntax]

```
<samples9298996>
  rdf:type bf:Text ;
  rdf:type bf:Work ;
  bf:title "Oliver Twist." ;
  bf:class <id.loc.gov/…/PZ3> ;
  bf:creator [
    rdf:type bf:Person ;
    bf:label "Dickens, Charles, 1812-1870." ;
  ] .

<id.loc.gov/…/PZ3>
  rdf:type bf:LCC ;
  bf:label "PZ3.D55O165PR4567" .
```

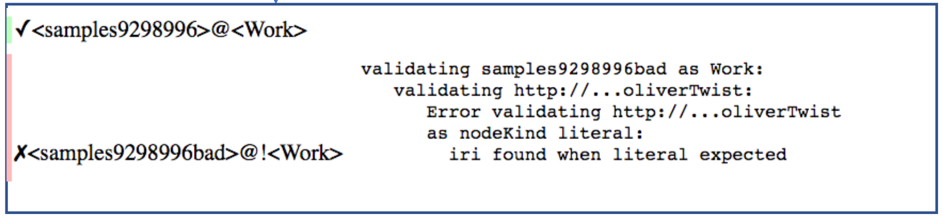


2. Against a schema.

1. Validate data.

A resource matching the “Classification” shape:
is identified with an IRI starting with “id.loc.gov”
has zero or one “rdf:type bf:LOC statements
optionally has an additional “rdf:type” statement
has exactly one “bf:label” statement with a literal value

3. To yield a validation result



```
✓<samples9298996>@<Work>
                                   validating samples9298996bad as Work:
                                     validating http://...oliverTwist:
                                       Error validating http://...oliverTwist
                                       as nodeKind literal:
✗<samples9298996bad>@!<Work>             iri found when literal expected
```

ShEx Schema [ShExC syntax]

```
<Work> EXTRA rdf:type {
  rdf:type [bf:Work] ? ;
  bf:title LITERAL ;
  bf:class @<Classification> * ;
  bf:creator @<Person> OR @<Organization> + ;
  bf:derivedFrom IRI * ;
}

<Classification>
  [<http://id.loc.gov/…/>~]
AND
  EXTRA rdf:type {
    rdf:type [bf:LCC] ? ;
    bf:label LITERAL ;
}
```

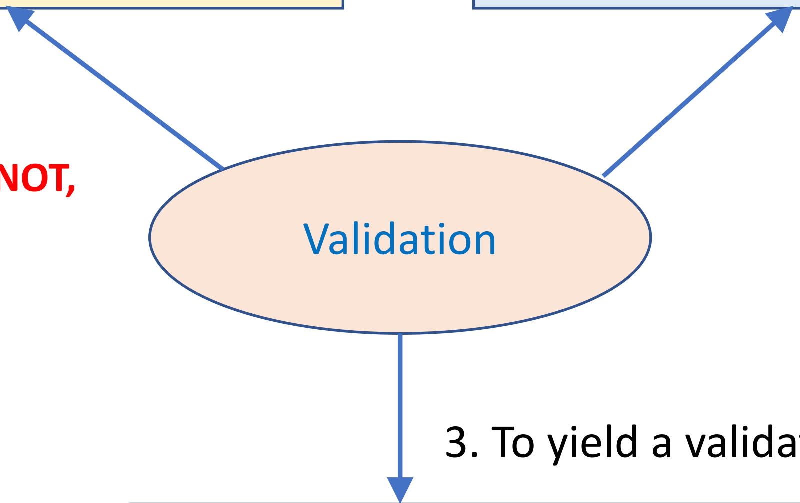
RDF Data [Turtle syntax]

```
<samples9298996>
  rdf:type bf:Text ;
  rdf:type bf:Work ;
  bf:title "Oliver Twist." ;
  bf:class <id.loc.gov/…/PZ3> ;
  bf:creator [
    rdf:type bf:Person ;
    bf:label "Dickens, Charles, 1812-1870." ;
  ] .

<id.loc.gov/…/PZ3>
  rdf:type bf:LCC ;
  bf:label "PZ3.D55O165PR4567" .
```

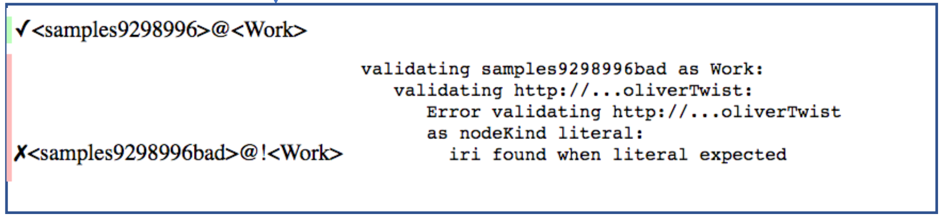
2. Against a schema.

1. Validate data.

Shapes can combine constraints with AND, OR, NOT, and parentheses.



3. To yield a validation result



```
✓<samples9298996>@<Work>
                                      validating samples9298996bad as Work:
                                         validating http://...oliverTwist:
                                            Error validating http://...oliverTwist
                                            as nodeKind literal:
✗<samples9298996bad>@!<Work>                  iri found when literal expected
```

Terminology summarized

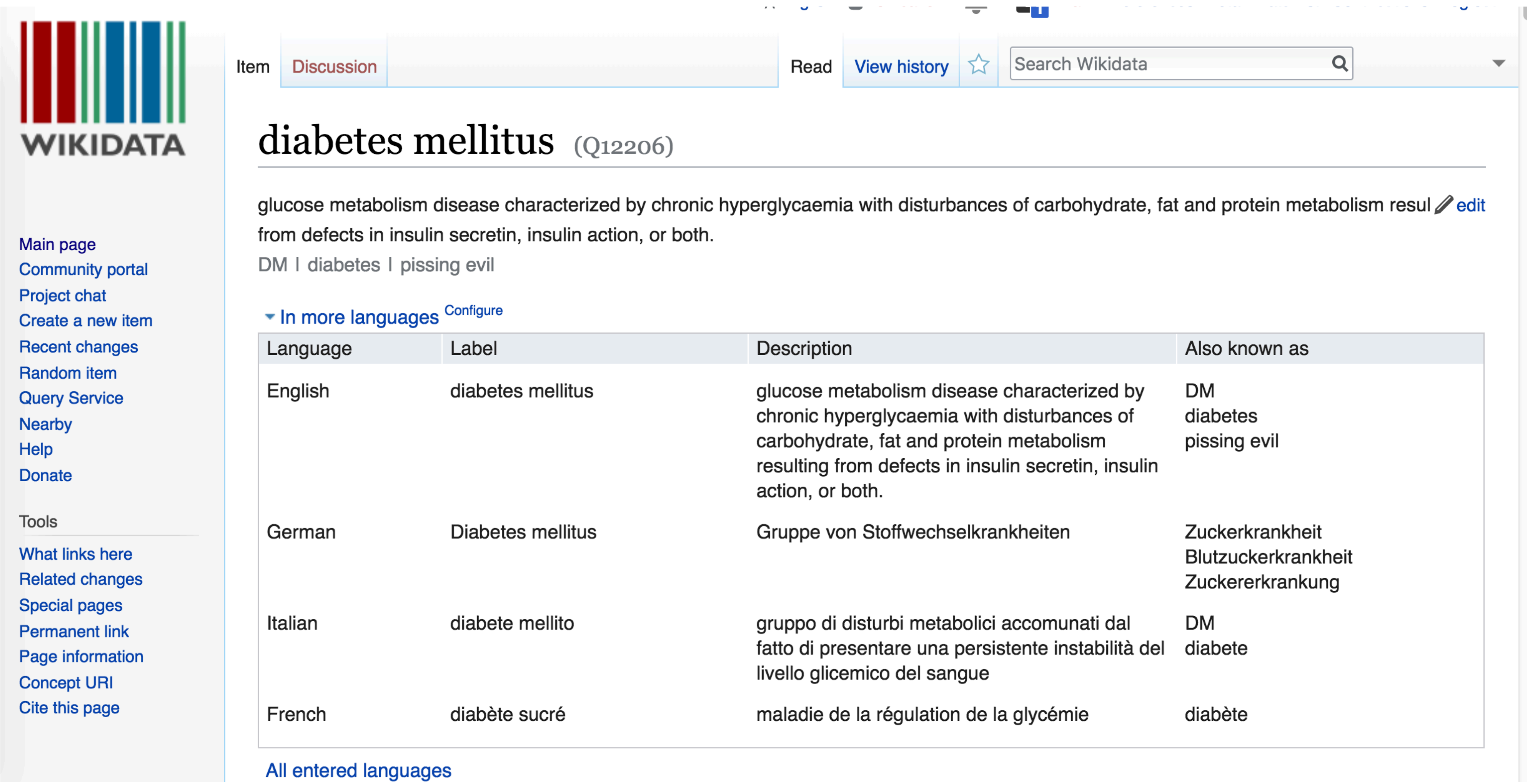
- A **ShEx Schema** holds a collection of...
  - **Shape Expressions**, which logically combine...
    - **Node Constraints**, which MUST match nodes in RDF Data, and...
    - **Shapes**, which MUST match triples in RDF Data as specified in...
      - **Triple constraints**, which MUST match triples touching a given node in RDF Data in terms of...
        - Predicates
        - Direction
        - Cardinality
        - Value

Example: Gene Wiki project

Try it

- Uses bots to synch resources on genes, proteins, diseases, drugs.
  - Due to open nature of Wikidata, described by different sets of properties.
- Role of ShEx [https://github.com/SuLab/Genewiki-ShEx]:
  1. Communicate underlying models used by bots.
  2. Capture errors and curation issues.

Wikidata page about a disease…



diabetes mellitus (Q12206)

glucose metabolism disease characterized by chronic hyperglycaemia with disturbances of carbohydrate, fat and protein metabolism resul from defects in insulin secretin, insulin action, or both.

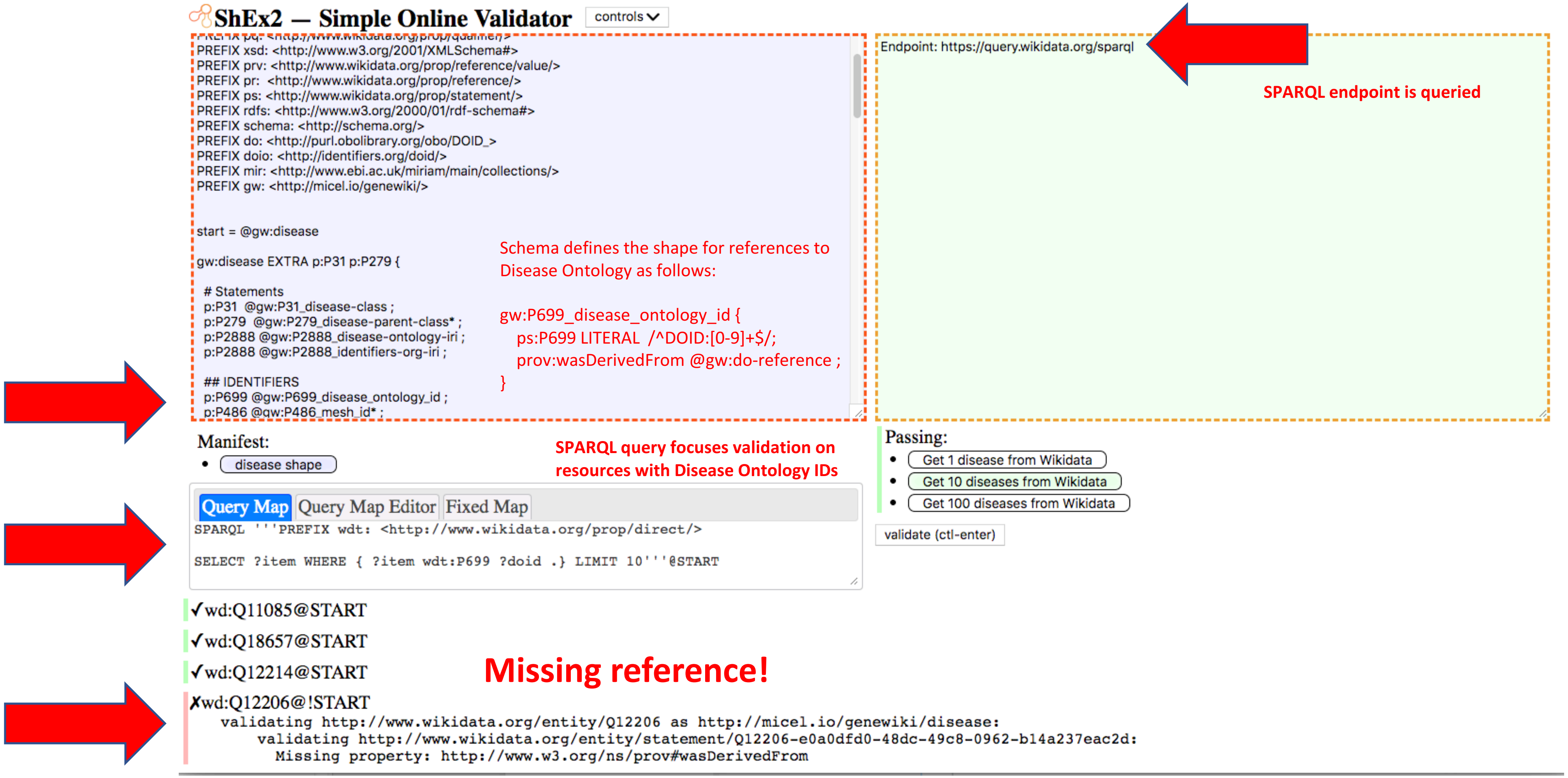
DM | diabetes | pissing evil

Language	Label	Description	Also known as
English	diabetes mellitus	glucose metabolism disease characterized by chronic hyperglycaemia with disturbances of carbohydrate, fat and protein metabolism resulting from defects in insulin secretin, insulin action, or both.	DM diabetes pissing evil
German	Diabetes mellitus	Gruppe von Stoffwechselkrankheiten	Zuckerkrankheit Blutzuckerkrankheit Zuckererkrankung
Italian	diabete mellito	gruppo di disturbi metabolici accomunati dal fatto di presentare una persistente instabilità del livello glicemico del sangue	DM diabete
French	diabète sucré	maladie de la régulation de la glycémie	diabète

https://www.wikidata.org/wiki/Q12206

…validated with ShEx

Try it



ShEx2 — Simple Online Validator

```
PREFIX pq: <http://www.wikidata.org/prop/qualifier/>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX prv: <http://www.wikidata.org/prop/reference/value/>
PREFIX pr:  <http://www.wikidata.org/prop/reference/>
PREFIX ps: <http://www.wikidata.org/prop/statement/>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX schema: <http://schema.org/>
PREFIX do: <http://purl.obolibrary.org/obo/DOID_>
PREFIX doio: <http://identifiers.org/doid/>
PREFIX mir: <http://www.ebi.ac.uk/miriam/main/collections/>
PREFIX gw: <http://micel.io/genewiki/>


start = @gw:disease

gw:disease EXTRA p:P31 p:P279 {

 # Statements
 p:P31  @gw:P31_disease-class ;
 p:P279  @gw:P279_disease-parent-class* ;
 p:P2888 @gw:P2888_disease-ontology-iri ;
 p:P2888 @gw:P2888_identifiers-org-iri ;

 ## IDENTIFIERS
 p:P699 @gw:P699_disease_ontology_id ;
 p:P486 @gw:P486_mesh_id* ;
```

Schema defines the shape for references to Disease Ontology as follows:

```
gw:P699_disease_ontology_id {
   ps:P699 LITERAL  /^DOID:[0-9]+$/;
   prov:wasDerivedFrom @gw:do-reference ;
}
```

Endpoint: https://query.wikidata.org/sparql

SPARQL endpoint is queried

Manifest:

- disease shape

SPARQL query focuses validation on resources with Disease Ontology IDs

Passing:

- Get 1 disease from Wikidata
- Get 10 diseases from Wikidata
- Get 100 diseases from Wikidata

```
SPARQL '''PREFIX wdt: <http://www.wikidata.org/prop/direct/>

SELECT ?item WHERE { ?item wdt:P699 ?doid .} LIMIT 10'''@START
```

✓wd:Q11085@START

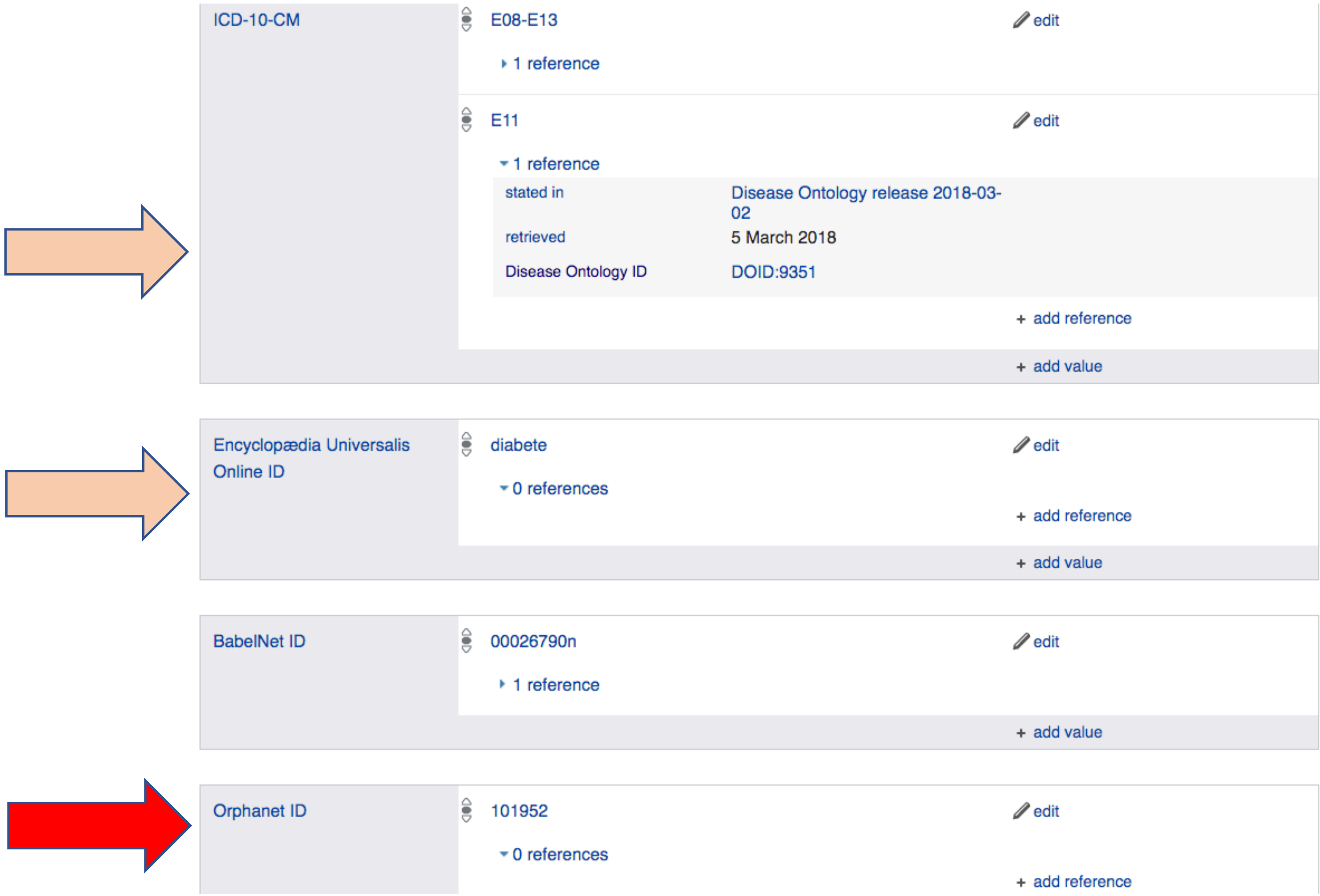
✓wd:Q18657@START

✓wd:Q12214@START

Missing reference!

```
✗wd:Q12206@!START
   validating http://www.wikidata.org/entity/Q12206 as http://micel.io/genewiki/disease:
       validating http://www.wikidata.org/entity/statement/Q12206-e0a0dfd0-48dc-49c8-0962-b14a237eac2d:
         Missing property: http://www.w3.org/ns/prov#wasDerivedFrom
```

…validated with ShEx



ICD-10-CM

E08-E13 ✎ edit

▸ 1 reference

E11 ✎ edit

▾ 1 reference

stated in	Disease Ontology release 2018-03-02
retrieved	5 March 2018
Disease Ontology ID	DOID:9351

+ add reference

+ add value

Encyclopædia Universalis Online ID

diabete ✎ edit

▾ 0 references

+ add reference

+ add value

BabelNet ID

00026790n ✎ edit

▸ 1 reference

+ add value

Orphanet ID

101952 ✎ edit

▾ 0 references

+ add reference

Reference done correctly

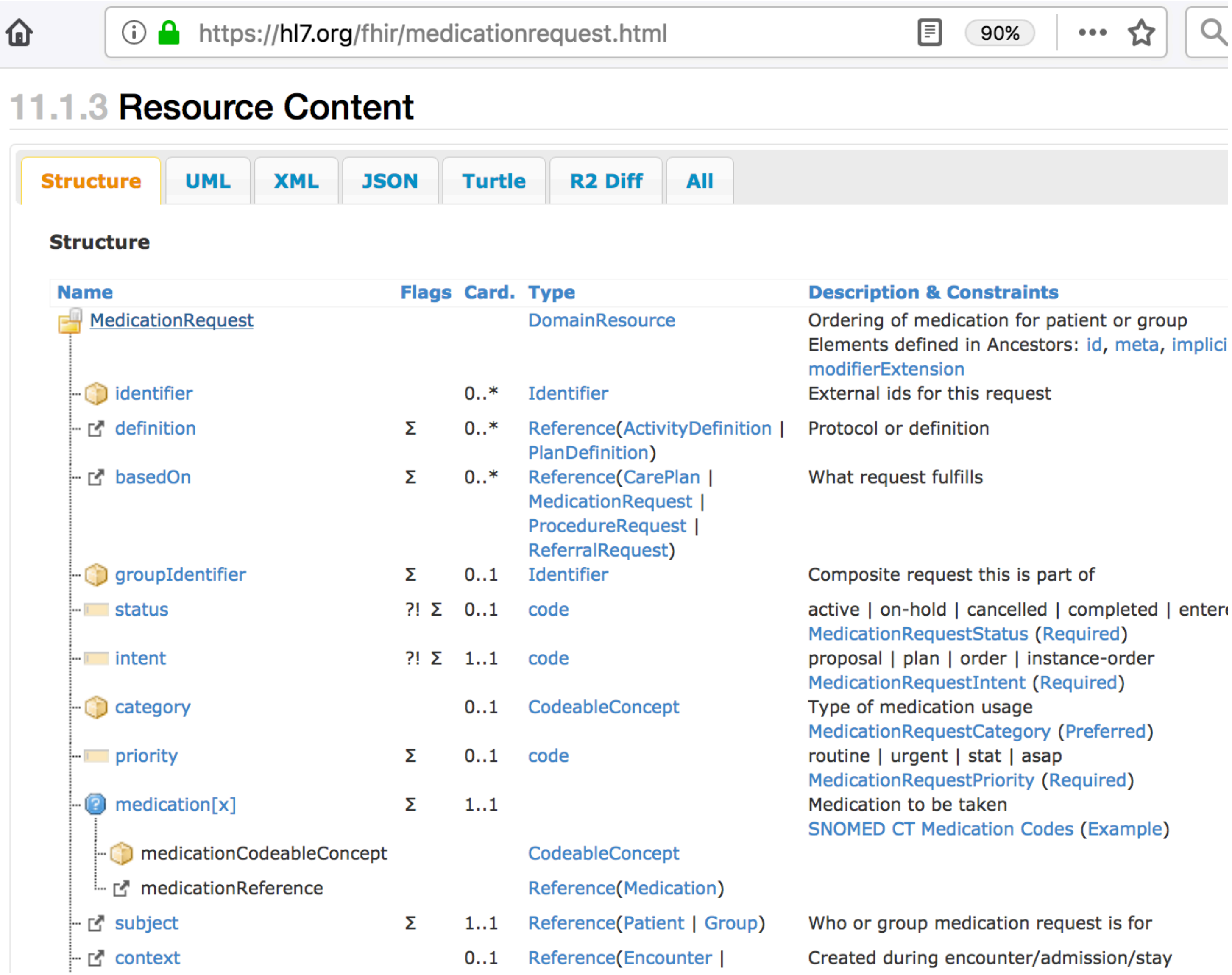
Reference not tested because ID did not match specified pattern

Reference missing on mapping to ID that matches specified pattern

Must be flagged to curation team:

- Add to Disease Ontology?
- Fix in Wikidata?
- Tweak ShEx to accept as is?

Example: Medication Request Definition (HL7 FHIR[1])



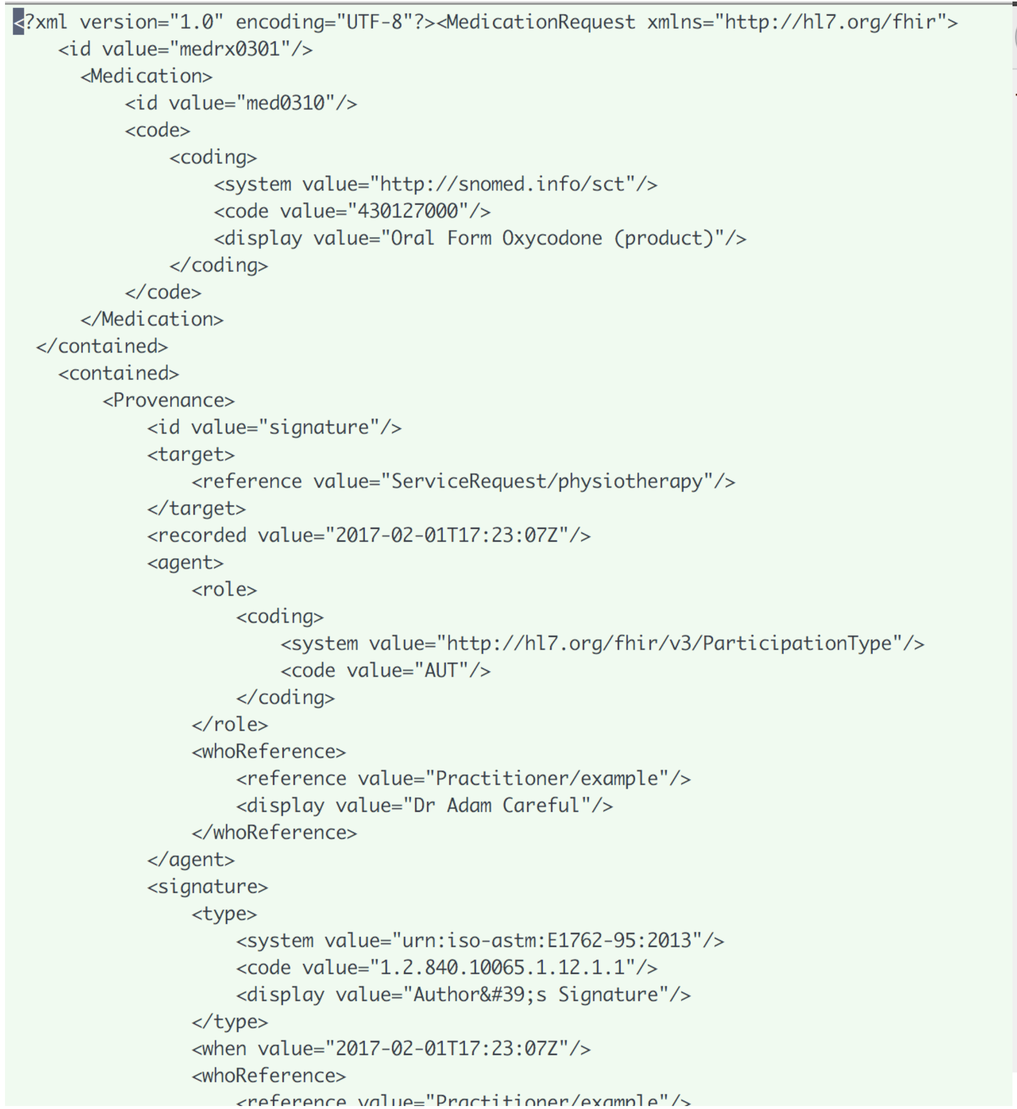
11.1.3 Resource Content

Structure

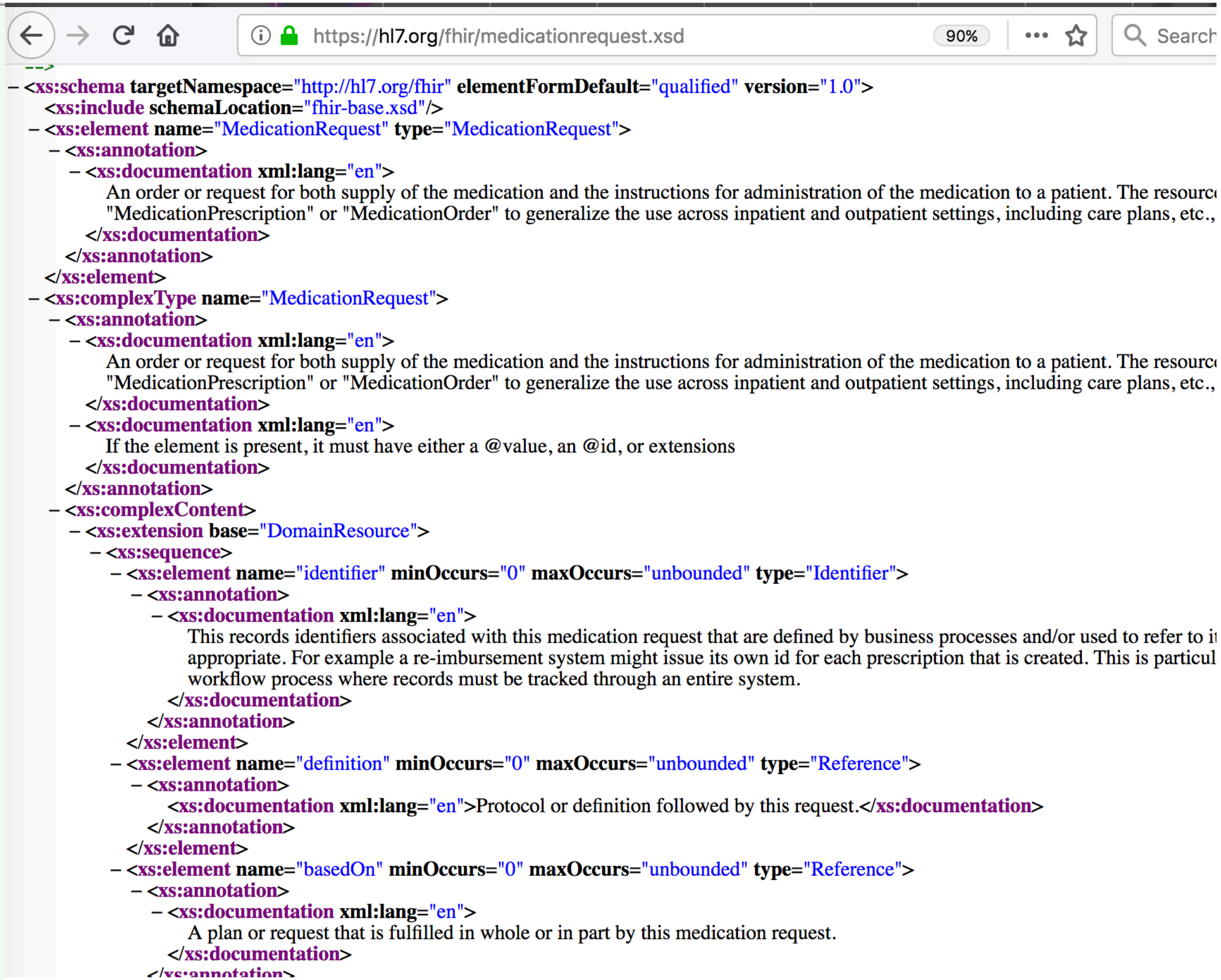
Name	Flags	Card.	Type	Description & Constraints
MedicationRequest			DomainResource	Ordering of medication for patient or group Elements defined in Ancestors: id, meta, implici modifierExtension
identifier		0..*	Identifier	External ids for this request
definition	Σ	0..*	Reference(ActivityDefinition | PlanDefinition)	Protocol or definition
basedOn	Σ	0..*	Reference(CarePlan | MedicationRequest | ProcedureRequest | ReferralRequest)	What request fulfills
groupIdentifier	Σ	0..1	Identifier	Composite request this is part of
status	?! Σ	0..1	code	active | on-hold | cancelled | completed | enter MedicationRequestStatus (Required)
intent	?! Σ	1..1	code	proposal | plan | order | instance-order MedicationRequestIntent (Required)
category		0..1	CodeableConcept	Type of medication usage MedicationRequestCategory (Preferred)
priority	Σ	0..1	code	routine | urgent | stat | asap MedicationRequestPriority (Required)
medication[x]	Σ	1..1		Medication to be taken SNOMED CT Medication Codes (Example)
medicationCodeableConcept			CodeableConcept	
medicationReference			Reference(Medication)	
subject	Σ	1..1	Reference(Patient | Group)	Who or group medication request is for
context		0..1	Reference(Encounter |	Created during encounter/admission/stay

[1] Fast Healthcare Interoperability Resources (FHIR), a standards framework of Health Level Seven International (HL7) for the exchange, integration, sharing, and retrieval of electronic health information

Medication Request [XML and XML Schema]

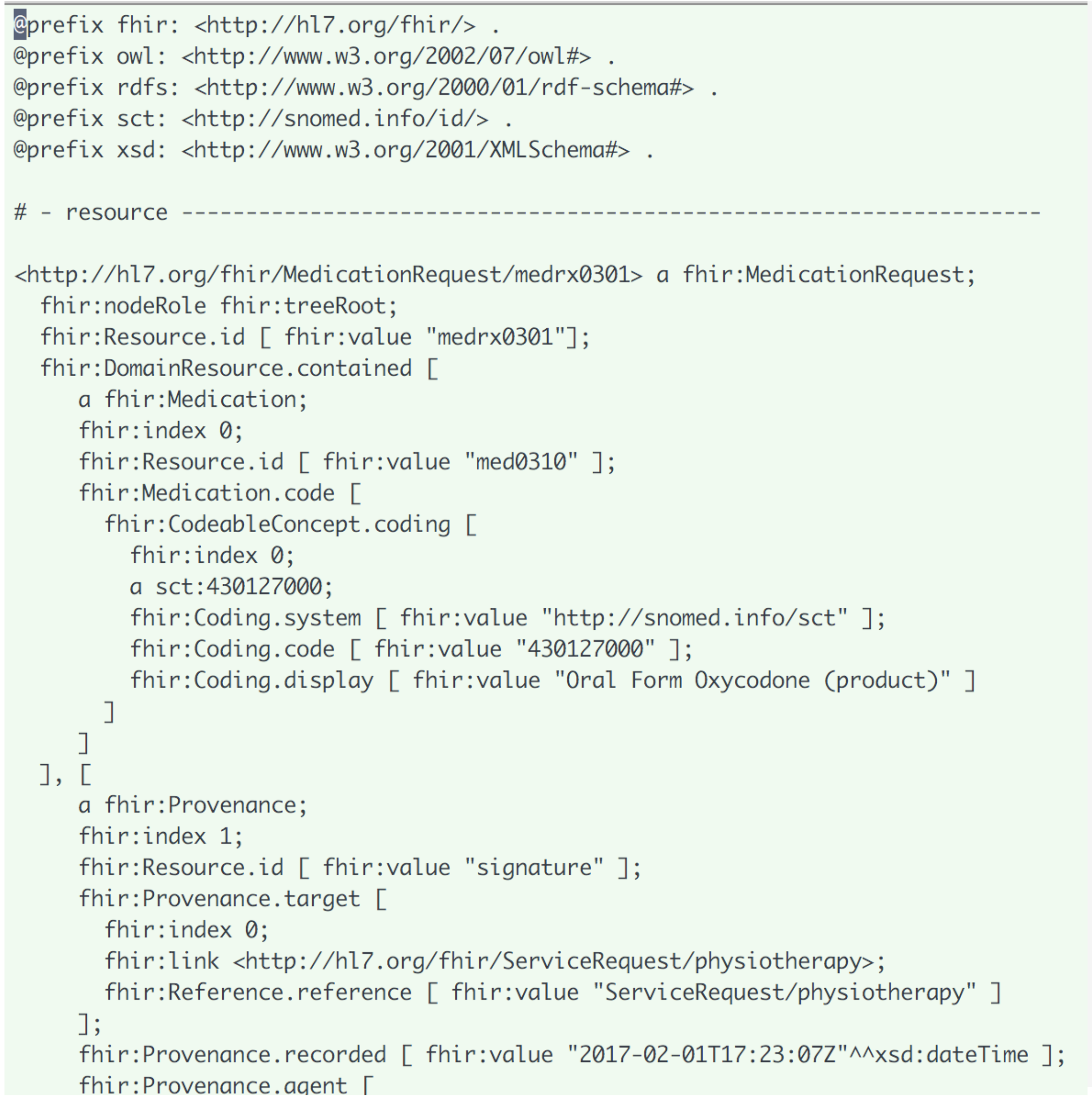


```
<?xml version="1.0" encoding="UTF-8"?><MedicationRequest xmlns="http://hl7.org/fhir">
    <id value="medrx0301"/>
      <Medication>
          <id value="med0310"/>
          <code>
              <coding>
                  <system value="http://snomed.info/sct"/>
                  <code value="430127000"/>
                  <display value="Oral Form Oxycodone (product)"/>
              </coding>
          </code>
      </Medication>
  </contained>
    <contained>
        <Provenance>
            <id value="signature"/>
            <target>
                <reference value="ServiceRequest/physiotherapy"/>
            </target>
            <recorded value="2017-02-01T17:23:07Z"/>
            <agent>
                <role>
                    <coding>
                        <system value="http://hl7.org/fhir/v3/ParticipationType"/>
                        <code value="AUT"/>
                    </coding>
                </role>
                <whoReference>
                    <reference value="Practitioner/example"/>
                    <display value="Dr Adam Careful"/>
                </whoReference>
            </agent>
            <signature>
                <type>
                    <system value="urn:iso-astm:E1762-95:2013"/>
                    <code value="1.2.840.10065.1.12.1.1"/>
                    <display value="Author&#39;s Signature"/>
                </type>
                <when value="2017-02-01T17:23:07Z"/>
                <whoReference>
                    <reference value="Practitioner/example"/>
```

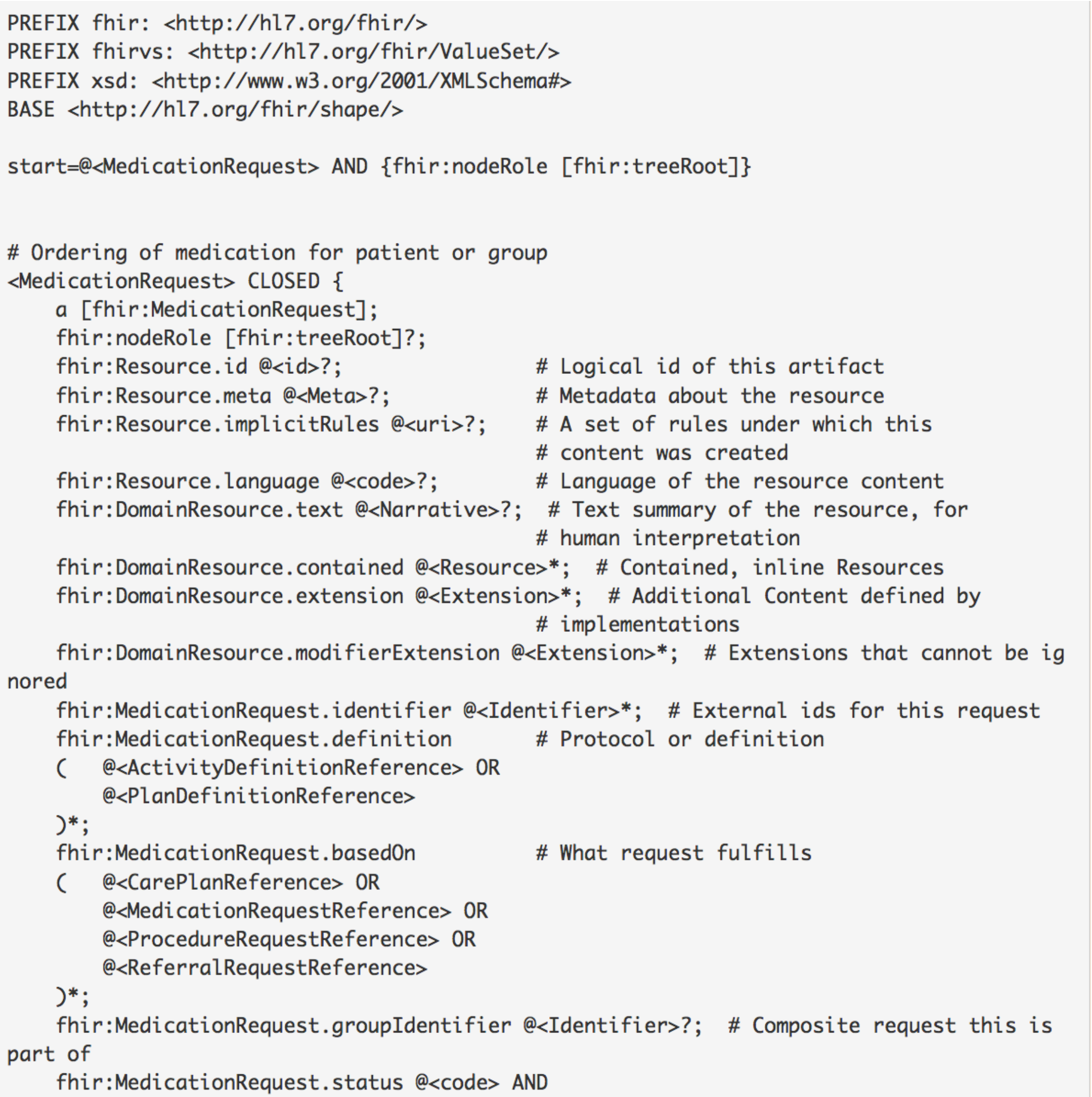


```
-->
– <xs:schema targetNamespace="http://hl7.org/fhir" elementFormDefault="qualified" version="1.0">
    <xs:include schemaLocation="fhir-base.xsd"/>
  – <xs:element name="MedicationRequest" type="MedicationRequest">
    – <xs:annotation>
      – <xs:documentation xml:lang="en">
          An order or request for both supply of the medication and the instructions for administration of the medication to a patient. The resourc
          "MedicationPrescription" or "MedicationOrder" to generalize the use across inpatient and outpatient settings, including care plans, etc.,
        </xs:documentation>
      </xs:annotation>
    </xs:element>
  – <xs:complexType name="MedicationRequest">
    – <xs:annotation>
      – <xs:documentation xml:lang="en">
          An order or request for both supply of the medication and the instructions for administration of the medication to a patient. The resourc
          "MedicationPrescription" or "MedicationOrder" to generalize the use across inpatient and outpatient settings, including care plans, etc.,
        </xs:documentation>
      – <xs:documentation xml:lang="en">
          If the element is present, it must have either a @value, an @id, or extensions
        </xs:documentation>
      </xs:annotation>
    – <xs:complexContent>
      – <xs:extension base="DomainResource">
        – <xs:sequence>
          – <xs:element name="identifier" minOccurs="0" maxOccurs="unbounded" type="Identifier">
            – <xs:annotation>
              – <xs:documentation xml:lang="en">
                  This records identifiers associated with this medication request that are defined by business processes and/or used to refer to i
                  appropriate. For example a re-imbursement system might issue its own id for each prescription that is created. This is particul
                  workflow process where records must be tracked through an entire system.
                </xs:documentation>
              </xs:annotation>
            </xs:element>
          – <xs:element name="definition" minOccurs="0" maxOccurs="unbounded" type="Reference">
            – <xs:annotation>
                <xs:documentation xml:lang="en">Protocol or definition followed by this request.</xs:documentation>
              </xs:annotation>
            </xs:element>
          – <xs:element name="basedOn" minOccurs="0" maxOccurs="unbounded" type="Reference">
            – <xs:annotation>
              – <xs:documentation xml:lang="en">
                  A plan or request that is fulfilled in whole or in part by this medication request.
                </xs:documentation>
              </xs:annotation>
```

Medication Request [RDF and ShEx]



```
@prefix fhir: <http://hl7.org/fhir/> .
@prefix owl: <http://www.w3.org/2002/07/owl#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix sct: <http://snomed.info/id/> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .

# - resource -------------------------------------------------------------------

<http://hl7.org/fhir/MedicationRequest/medrx0301> a fhir:MedicationRequest;
  fhir:nodeRole fhir:treeRoot;
  fhir:Resource.id [ fhir:value "medrx0301"];
  fhir:DomainResource.contained [
     a fhir:Medication;
     fhir:index 0;
     fhir:Resource.id [ fhir:value "med0310" ];
     fhir:Medication.code [
       fhir:CodeableConcept.coding [
         fhir:index 0;
         a sct:430127000;
         fhir:Coding.system [ fhir:value "http://snomed.info/sct" ];
         fhir:Coding.code [ fhir:value "430127000" ];
         fhir:Coding.display [ fhir:value "Oral Form Oxycodone (product)" ]
       ]
     ]
  ], [
     a fhir:Provenance;
     fhir:index 1;
     fhir:Resource.id [ fhir:value "signature" ];
     fhir:Provenance.target [
       fhir:index 0;
       fhir:link <http://hl7.org/fhir/ServiceRequest/physiotherapy>;
       fhir:Reference.reference [ fhir:value "ServiceRequest/physiotherapy" ]
     ];
     fhir:Provenance.recorded [ fhir:value "2017-02-01T17:23:07Z"^^xsd:dateTime ];
     fhir:Provenance.agent [
```



```
PREFIX fhir: <http://hl7.org/fhir/>
PREFIX fhirvs: <http://hl7.org/fhir/ValueSet/>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
BASE <http://hl7.org/fhir/shape/>

start=@<MedicationRequest> AND {fhir:nodeRole [fhir:treeRoot]}


# Ordering of medication for patient or group
<MedicationRequest> CLOSED {
    a [fhir:MedicationRequest];
    fhir:nodeRole [fhir:treeRoot]?;
    fhir:Resource.id @<id>?;                # Logical id of this artifact
    fhir:Resource.meta @<Meta>?;            # Metadata about the resource
    fhir:Resource.implicitRules @<uri>?;    # A set of rules under which this
                                            # content was created
    fhir:Resource.language @<code>?;        # Language of the resource content
    fhir:DomainResource.text @<Narrative>?;  # Text summary of the resource, for
                                            # human interpretation
    fhir:DomainResource.contained @<Resource>*;  # Contained, inline Resources
    fhir:DomainResource.extension @<Extension>*;  # Additional Content defined by
                                            # implementations
    fhir:DomainResource.modifierExtension @<Extension>*;  # Extensions that cannot be ig
nored
    fhir:MedicationRequest.identifier @<Identifier>*;  # External ids for this request
    fhir:MedicationRequest.definition       # Protocol or definition
    (   @<ActivityDefinitionReference> OR
        @<PlanDefinitionReference>
    )*;
    fhir:MedicationRequest.basedOn          # What request fulfills
    (   @<CarePlanReference> OR
        @<MedicationRequestReference> OR
        @<ProcedureRequestReference> OR
        @<ReferralRequestReference>
    )*;
    fhir:MedicationRequest.groupIdentifier @<Identifier>?;  # Composite request this is
part of
    fhir:MedicationRequest.status @<code> AND
```

Open-source implementations

Javascript	passes all 1,077 tests
Scala	passes all
Python	passes all for ShEx 2.0, working on 2.1
Ruby	passes most
Java	passes most

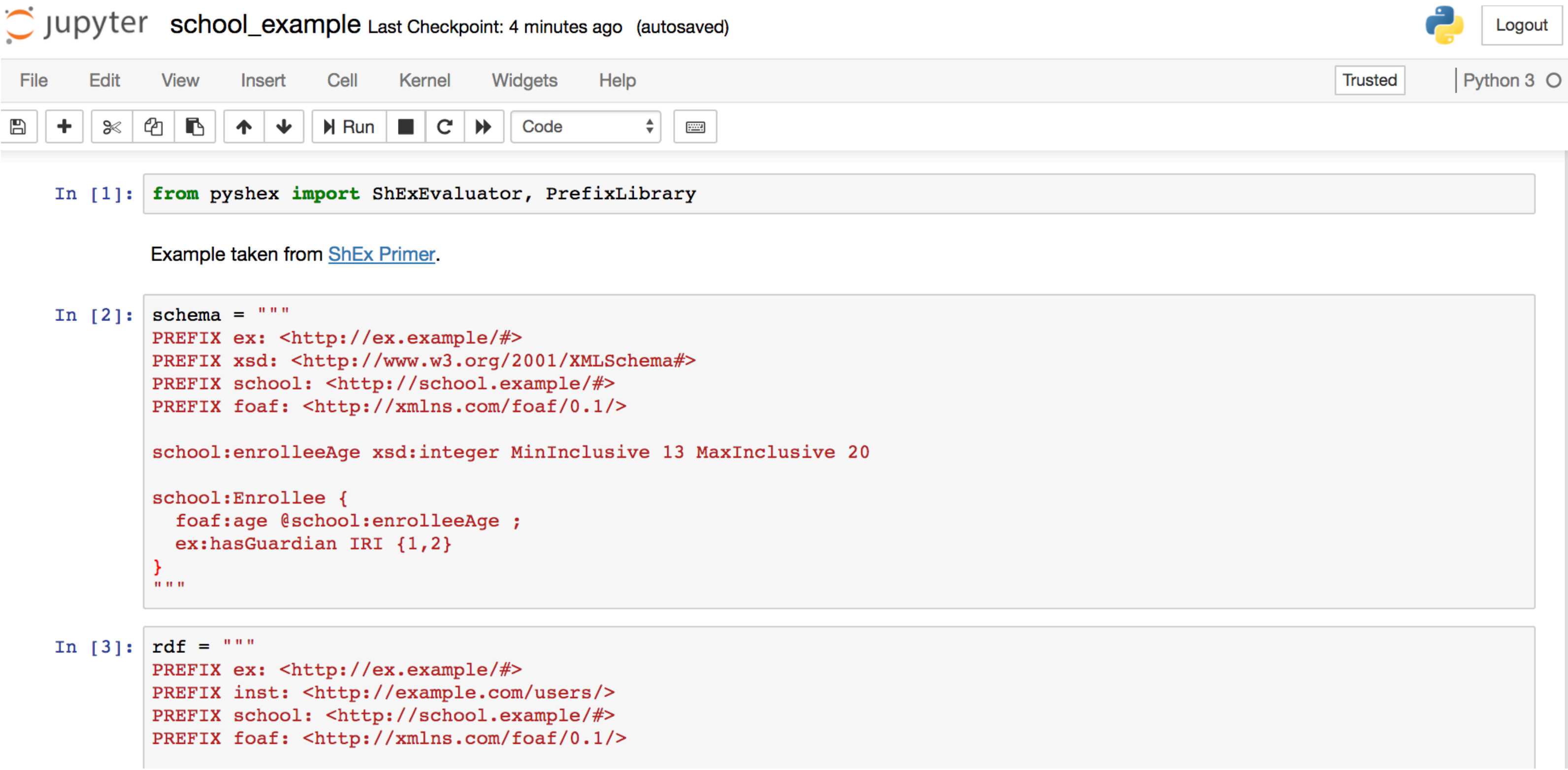
ShEx Online Validator

- https://rawgit.com/shexSpec/shex.js/master/doc/shex-simple.html?manifestURL=https://www.w3.org/2017/10/bibframe-shex/shex-simple-examples.json
- http://bit.ly/work_has_one_title
- http://bit.ly/work_has_type_arc
- http://bit.ly/shape_references_another_shape
- http://bit.ly/shape_references_an_authority
- http://bit.ly/shape_combines_constraints
- http://bit.ly/shape_references_alternative_shapes
- http://bit.ly/disease_ontology_references

Jupyter Notebooks

- Jupyter Notebook
  - Interactive, browser-based environment mixing code, graphics, and rich text.
  - Work savable as JSON, HTML, LaTeX, PDF, HTML slideshows, Markdown, or executable code.
  - Used to log the process of data exploration, document results, even author books.
  - Recent trends: mixed-language notebooks, code completion (and other IDE-like features).
- ShEx in Jupyter Notebook
  - Work in progress – see https://github.com/hsolbrig/PyShEx/blob/master/notebooks
  - Goals
    - Cell magic or drop-downs for flagging ShEx schemas, RDF data, shape maps.
    - ShEx Primer as a Jupyter notebook.

Jupyter Notebooks



```
In [1]: from pyshex import ShExEvaluator, PrefixLibrary
```

Example taken from ShEx Primer.

```
In [2]: schema = """
        PREFIX ex: <http://ex.example/#>
        PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
        PREFIX school: <http://school.example/#>
        PREFIX foaf: <http://xmlns.com/foaf/0.1/>

        school:enrolleeAge xsd:integer MinInclusive 13 MaxInclusive 20

        school:Enrollee {
          foaf:age @school:enrolleeAge ;
          ex:hasGuardian IRI {1,2}
        }
        """
```

```
In [3]: rdf = """
        PREFIX ex: <http://ex.example/#>
        PREFIX inst: <http://example.com/users/>
        PREFIX school: <http://school.example/#>
        PREFIX foaf: <http://xmlns.com/foaf/0.1/>
```

Active and friendly development community

- W3C Shape Expressions Community Group (ShEx CG)
    - https://www.w3.org/community/shex CG homepage
    - http://shex.io ShEx homepage
    - http://shex.io/shex-primer/ primer
    - http://shex.io/shex-semantics/ main specification
    - https://rawgit.com/shexSpec/shex.js/master/doc/shex-simple.html validator