Framing Improvements (Value Patterns)

Exact Value Match

```
{
  "@context": {"ex": "http://example.org/"},
  "@id": "ex:Sub1",
  "ex:p": "P",
  "ex:q": {"@value": "Q", "@type": "ex:q"},
  "ex:r": {"@value": "R", "@language": "r"}
}
```

Match on exact type/ language but wildcard value

```
{
  "@context": {"ex": "http://example.org/"},
  "ex:p": {"@value": {}},
  "ex:q": {"@value": {}, "@type": "ex:q"},
  "ex:r": {"@value": {}, "@language": "r"}
}
```

Match on exact value wildcard type/language

```
{
  "@context": {"ex": "http://example.org/"},
  "ex:q": {"@value": "Q", "@type": {}}
}
```

Match on no value

```
{
  "@context": {"ex": "http://example.org/"},
  "ex:p": {"@value": {}, "@type": []},
  "ex:q": {"@value": {}, "@type": "ex:q"},
  "ex:r": {"@value": {}, "@language": "r"}
}
```

- Also match on any selected value/type/language with […]

Future Directions

- JSON-LD* – annotating triples in JSON-LD [1]
- Logic programming JSON-LD – Towards abstracting logic from the Notation-3 syntax.
    - Formulae, Variables, Implication, operations on Lists.

[1] https://json-ld.github.io/json-ld-star/