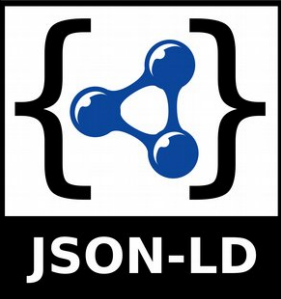


JSON-LD

Aurelijus Banelis



is W3C standard



JSON-LD 1.0

A JSON-based Serialization for Linked Data

W3C Recommendation 16 January 2014

This version:

<http://www.w3.org/TR/2014/REC-json-ld-20140116/>

Latest published version:

<http://www.w3.org/TR/json-ld/>

Previous version:

<http://www.w3.org/TR/2013/PR-json-ld-20131105/>

Editors:

[Manu Sporny](#), [Digital Bazaar](#)

[Gregg Kellogg](#), [Kellogg Associates](#)

[Markus Lanthaler](#), [Graz University of Technology](#)

Authors:

[Manu Sporny](#), [Digital Bazaar](#)

[Dave Longley](#), [Digital Bazaar](#)

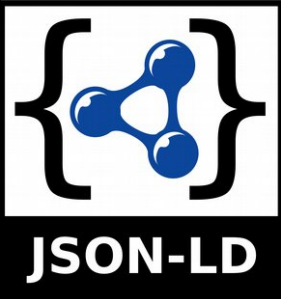
[Gregg Kellogg](#), [Kellogg Associates](#)

[Markus Lanthaler](#), [Graz University of Technology](#)

[Niklas Lindström](#)

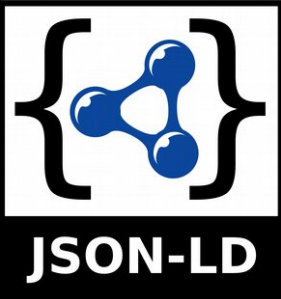
Examples

Specification

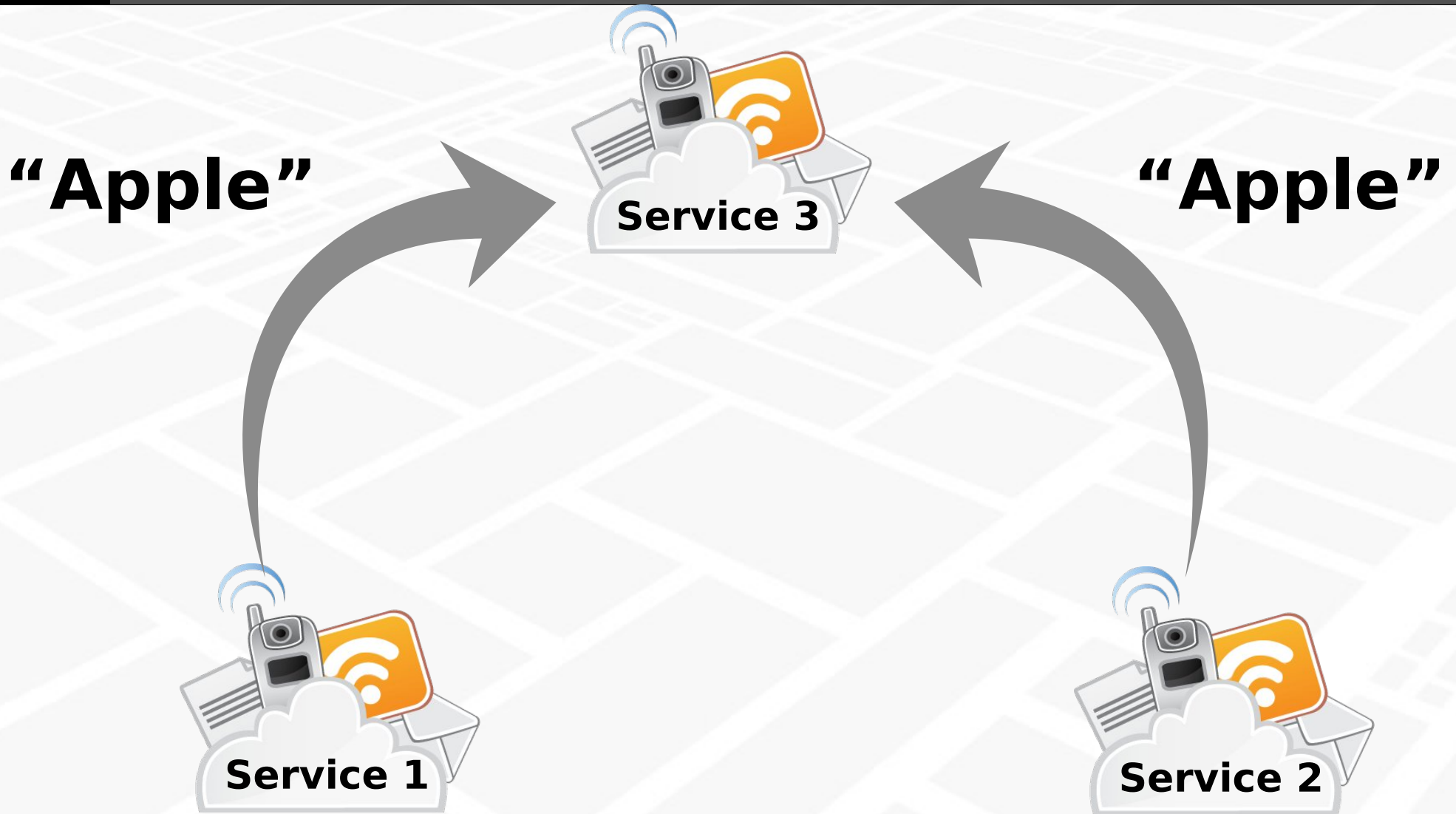


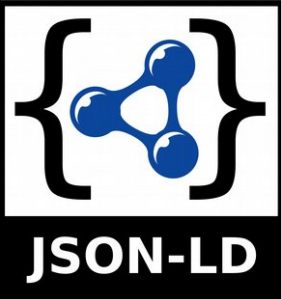
solves naming problem



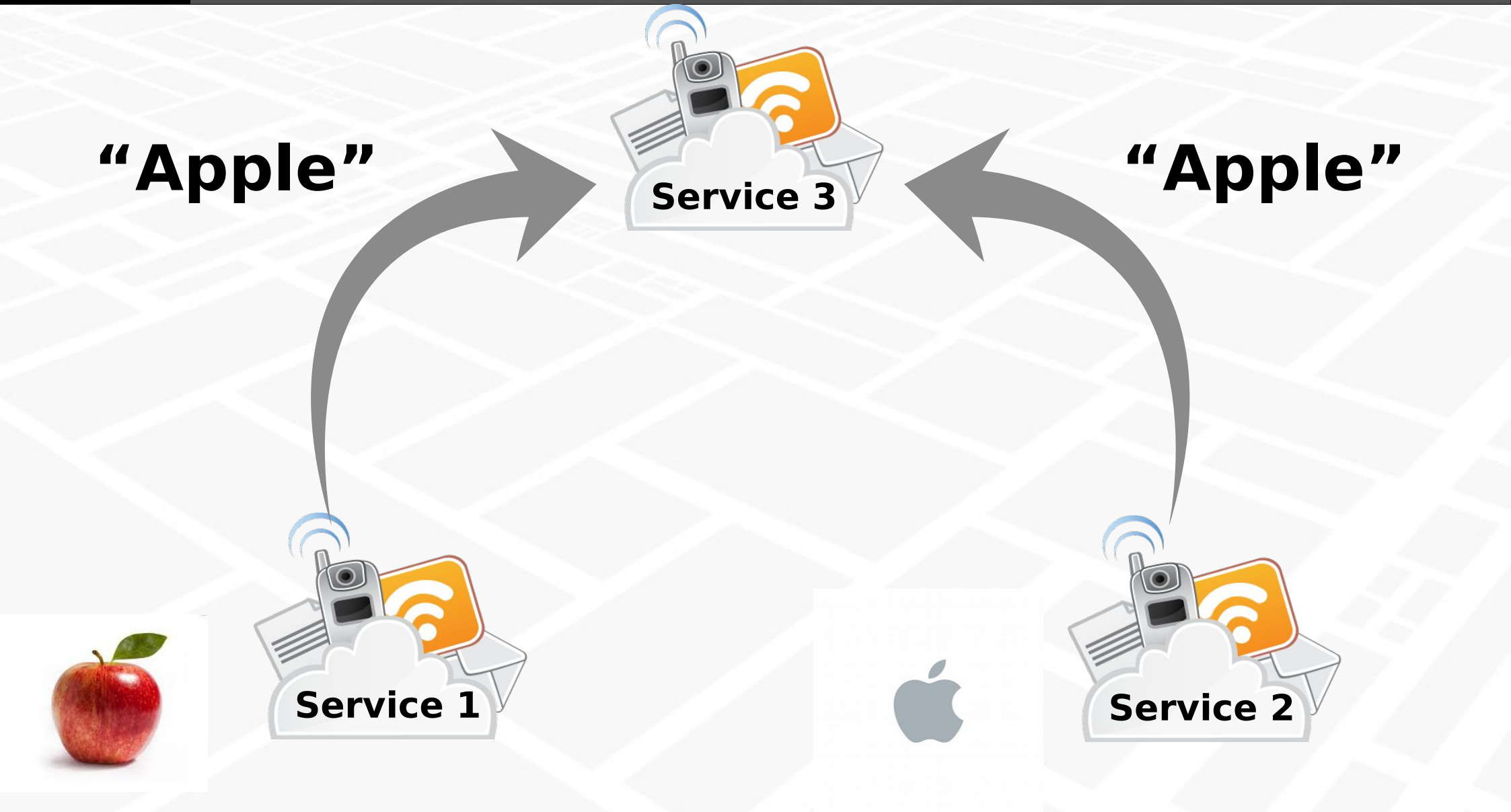


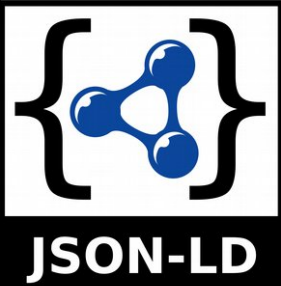
solves naming problem





solves naming problem

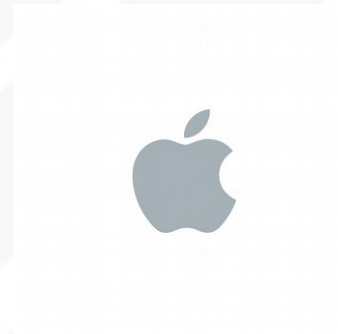


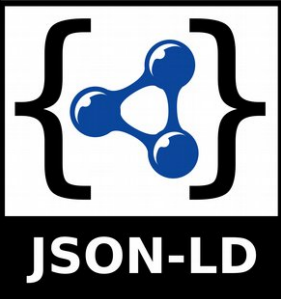


solves naming problem

**“http://
dreamatico.com/
data_images/
Apple/
apple-7.jpg”**

**“https://
www.apple.com/
v/home/
bq/images/
og.jpg”**





solves naming problem

“dc:apple-7.jpg”

“ac:og.jpg”

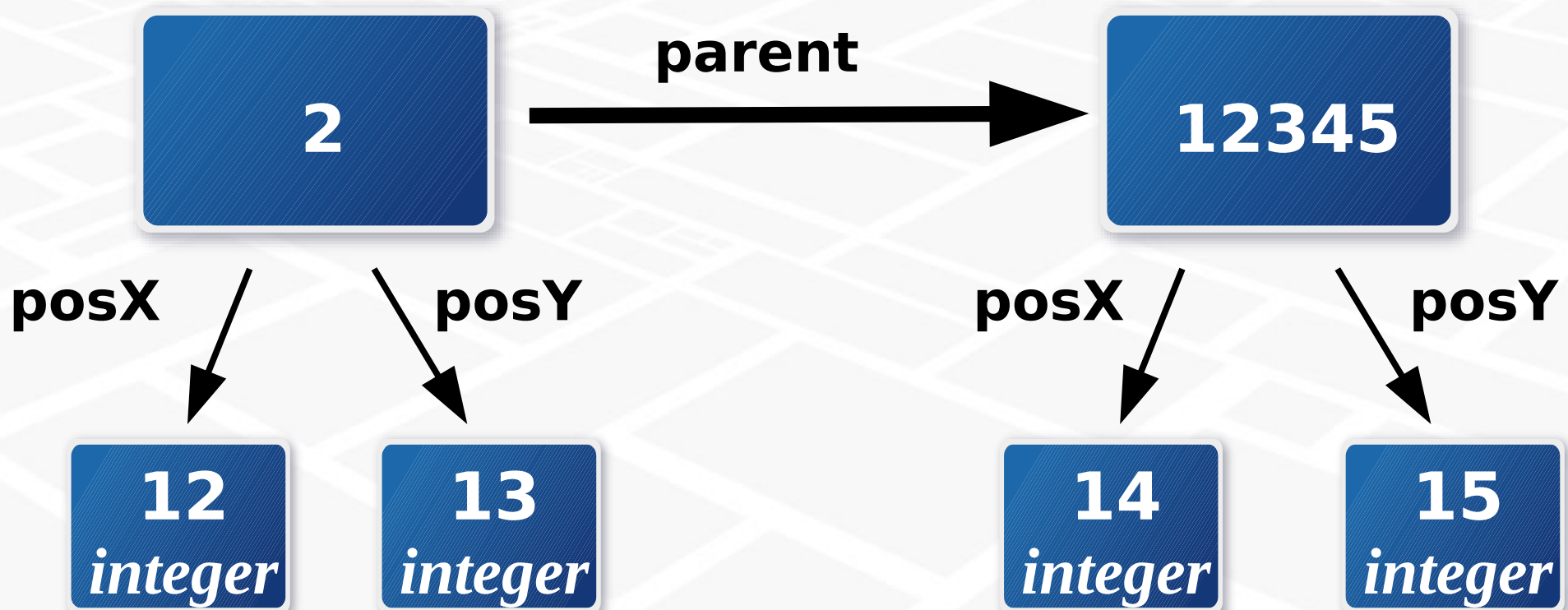
Service 3

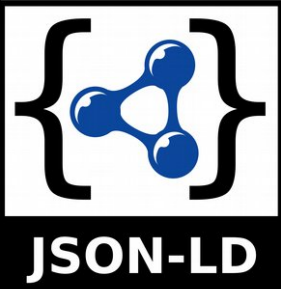
Service 1

Service 2



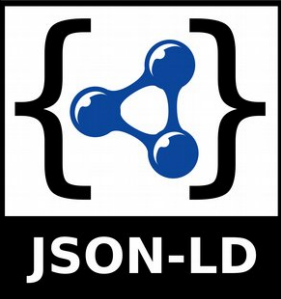
Linked data example





From simple JSON

```
{
  "nodes": [
    {
      "id": "12345",
      "posX": 12,
      "posY": 14,
      "parent": null
    },
    {
      "id": "2",
      "posX": 15,
      "posY": 16,
      "parent": "12345"
    }
  ]
}
```



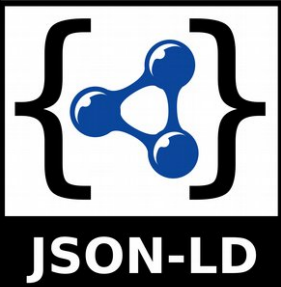
To JSON-LD

```
{
  "@context": "http://auginte.com/ns/s.jsonld",
  "nodes": [
    {
      "@id": "gn:12345",
      "posX": 12,
      "posY": 14,
      "parent": null
    },
    {
      "@id": "gn:2",
      "posX": 15,
      "posY": 16,
      "parent": "gn:12345"
    }
  ]
}
```

gn:

@id

@context



For computer

```
[{"http://auginte.com/ns/v0.6/node/list": [
  {"@id": "auginte://localhost/zooming/nodes/12345",
    "http://auginte.com/ns/v0.6/node/x": [
      {"@type": "http://www.w3.org/2001/XMLSchema#integer",
        "@value": 12}],
    "http://auginte.com/ns/v0.6/node/y": [
      {"@type": "http://www.w3.org/2001/XMLSchema#integer",
        "@value": 14
      }
    ]
  },
  {"@id": "auginte://localhost/zooming/nodes/2",
    "http://auginte.com/ns/v0.6/reference/node/parent": [
      {"@id": "auginte://localhost/zooming/nodes/12345"}
    ],
    "http://auginte.com/ns/v0.6/node/x": [
      {"@type": "http://www.w3.org/2001/XMLSchema#integer",
        "@value": 15}],
    "http://auginte.com/ns/v0.6/node/y": [
      {"@type": "http://www.w3.org/2001/XMLSchema#integer",
        "@value": 16
      }
    ]
  }
]}
```

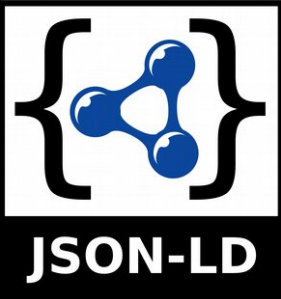
key

value

type

<http://json-ld.org/playground/index.html>

<https://github.com/aurelijusb/example-jsonld-php/>



Why and where (not) to use

*Personal opinion



Learn

JSON-LD
Playground



PHP

Libraries



**Semantic
WEB**

RDF
compatible



SEO

As microdata
(not all)

2nd fetch
for context

REST



OrientDB
export JSON

Storage



Scala.js
Prickle

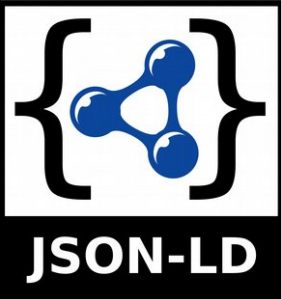
**Frontend
Backend**



Protobuf
Thrift

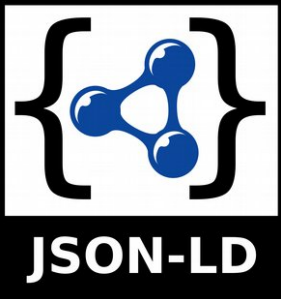
**Cross
language**





Conclusion

Do not reinvent the wheel
Use standards
For right job
There is no one-fits-all solution



Questions?

Do not reinvent the wheel

Use standards

For right job

There is no one-fits-all solution

Slides already at: aurelijus.banelis.lt

References and useful links

- <http://json-ld.org/>
- <https://github.com/aurelijusb/example-jsonld-php/>
- <https://github.com/lanthaler/JsonLD>
- <http://www.w3.org/TR/json-ld/>
- <https://developers.google.com/webmasters/business-location-pages/schema.org-examples>
- <https://schema.org>
- <https://developers.google.com/structured-data/testing-tool/>
- <https://github.com/json-ld/json-ld.org/wiki/Users-of-JSON-LD>
- <http://www.markus-lanthaler.com/hydra/>
-