

## Page 78

---

The link or citation is apparently broken.

### JSON-LD

A more modern way to pass data from one component to another in a Web application is using *JSON*, the Javascript Object Notation on citebray2014javascript. In order to make linked data in RDF more available to applications that use JSON, the W3C has recommended JSON-LD, JSON for Linked Data [Kellogg et al. 2014]. There is a direct correspondence between JSON-LD and RDF triples, making JSON-LD another serialization format for RDF.

## Page 84

---

- **Merging**—The process of treating two graphs as if they were one.
- **URI (Uniform Resource Indicator)**—A generalization of the URL (Uniform Resource Locator), which is the global name on the Web.
- **namespace**—A set of names that belongs to a single authority. Namespaces

Perhaps the word **Identifier** was meant, like in <https://tools.ietf.org/html/rfc3986>.

On the same page, I think the word None should not be there:

- **Reification**—The practice of making a statement  
It is done in RDF using
- **rdf:subject, rdf:predicate, and Nonerd:object**.
- **N-Triples, Turtle, RDF/XML**—The serialization syr

Also, maybe in the list of serialization formats on the next line, N3, TriG and N-Quads might also be mentioned.

## Page 103

---

ence it. When an agent accesses that HTTP URI, the server provides the page itself.

1. For added security, most web sites today use a more secure version of the HTTP protocol, called **HTTPS**. For the purposes of this book, when we refer to an “HTTP URI” we include URIs that use the HTTP protocol.

 Should say HTTPS, I believe

## Page 105

---

data, and we'll have a closer look at them later.

The reliance on URIs is key to distributed data, to the Web-based architecture, and to the integration of datasets coming from different sources and produced by different means. URIs allow us to make vocabulary and resource identification unambiguous by providing a global identification mechanism for the terms and subjects of our descriptions. URIs also have a *deferencing* mechanism, that is, a method that uses the URI to find a location on the Web from which further information can be loaded. Applications use this universal identification mechanism to ensure they are processing the right data in the right way and they use the dereferencing mechanism to obtain more data on demand.

When URIs are used and reused across RDF graphs and datasets, they provide junction points that allow us to merge the datasets regardless of their provenance, potentially forming a giant global graph. This extensible nature of RDF graphs provides a way to weave a world-wide Web of data.

I believe the word `deferencing` in the red rectangle should be `dereferencing`, just as it is written a few lines below.

## Page 111

---

These schemes have the advantage that they are stable; we can use the same URI for the HTML presentation of the latest RFC version, even as new versions are published. It has the disadvantage that I cannot refer to RDF 3968 itself, only to particular documents about it. This violates one of our guidelines for URI formation, that a URI should not include implementation-specific information about a document. We'll see later in this chapter how to use content negotiation to address this issue.

I guess it should say `RFC 3968`.

## Page 120

---

shown in Figure 5.4, which is available at the URL <https://dblp.org/rec/html/books/daglib/0028543>, and in machine-readable form in RDF, which is available from the URL <https://dblp.org/rec/rdf/books/daglib/0028543>. The triples for this part of the data sets are given in Turtle format in Figure 5.5.

```
http://www.bbc.co.uk/nature/life/Great_white_shark.rdf
```

DBLP is a simple example of how the same data can be served in different forms for different purposes; the human-readable HTML is difficult for a machine to read because of typesetting and styling which make it easier and more attractive for

This link seems out of its place. This section is about DBLP, but the link is about BBC and lacks any comments as to its meaning. In addition, it seems to lead to an HTTP 404 page.

## Page 150

---

from the example.

Since the example from the data graph matches the graph pattern triple for triple, we already know a lot about the graph pattern we want to create. The only thing we need to specify is which values from the example we want to keep as literal values in the pattern, and which ones we want to replace with question words. In Figure 6.7(a), the boxed x on certain resources (James Dean, John Ford, and Woman) indicates those that will stay as they are; all other resources (Natalie Wood, *The Searchers*, *Rebel Without a Cause*) will be replaced with question words. We have

I believe the diagram actually uses checkmark character ✓ to portray these items, not a boxed x.

## Page 217

- **Inference rules**—Systematic patterns defining which of the triples should be inferred.
- **Inference engine**—A program that performs inferences according to some inference rules. It is often integrated with a query engine.
- **SHACL shapes** to validate RDF data at the structural level by providing the constraint you expect them to meet.

I believe the words SHACL shapes should be in bold, just as the other terms in the **Fundamental concepts** list.

## Page 458

at all; there is no inference rule that tells us anything about Eddie.

What we really wanted to say was

```
epa:Eagle
  rdfs:subClassOf [a owl:Restriction ;
                  owl:onProperty epa:hasPart ;
                  owl:someValuesFrom epa:Wings ]
```

That is, if something is an Eagle, then it has parts that are wings.

How about the statement about being a subclass of EndangeredSpecies? The

## Page 462

modeling interpretation.

A fundamental example of this kind of clash can be found in examining the role of a class in a model. In object modeling, a class is basically a template from which an instance is stamped. It makes little or no sense to speak of multiple classes (stamped out of two templates?) or of having a property that isn't in the class (where do you put it if there wasn't a slot in the template for it?).

Multiple inheritance, where a subclass is derived from multiple classes, does exist in many programming languages, in Python for example. The object is considered an instance of every class in the inheritance hierarchy.

