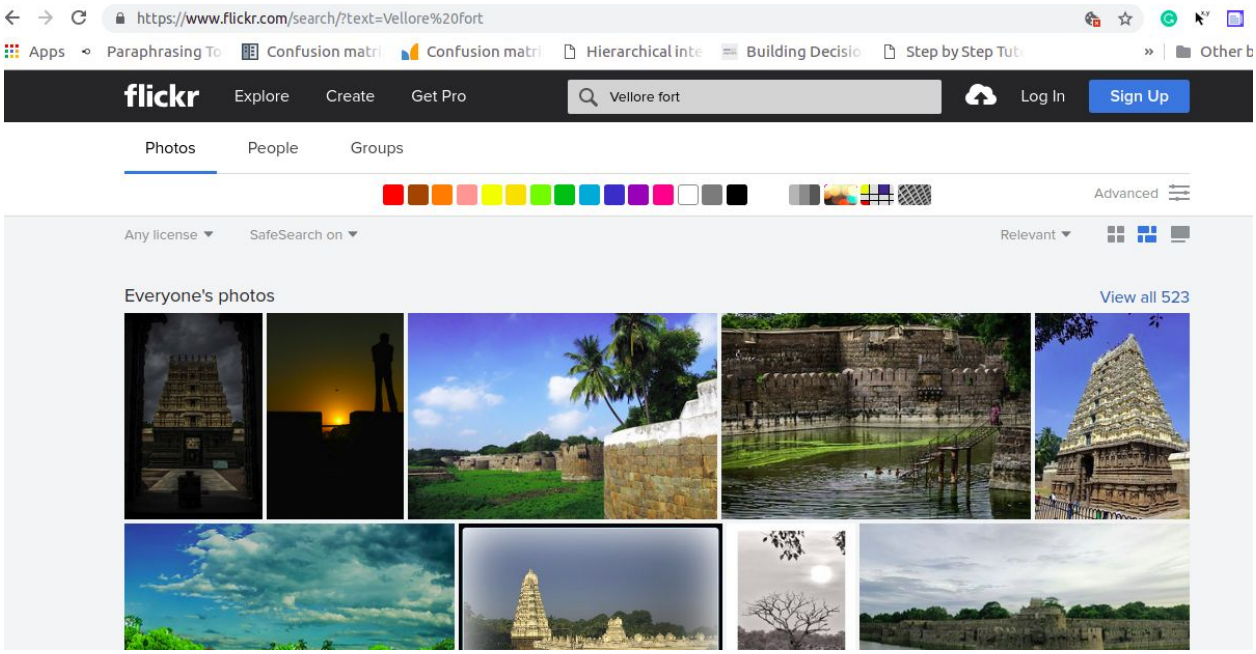
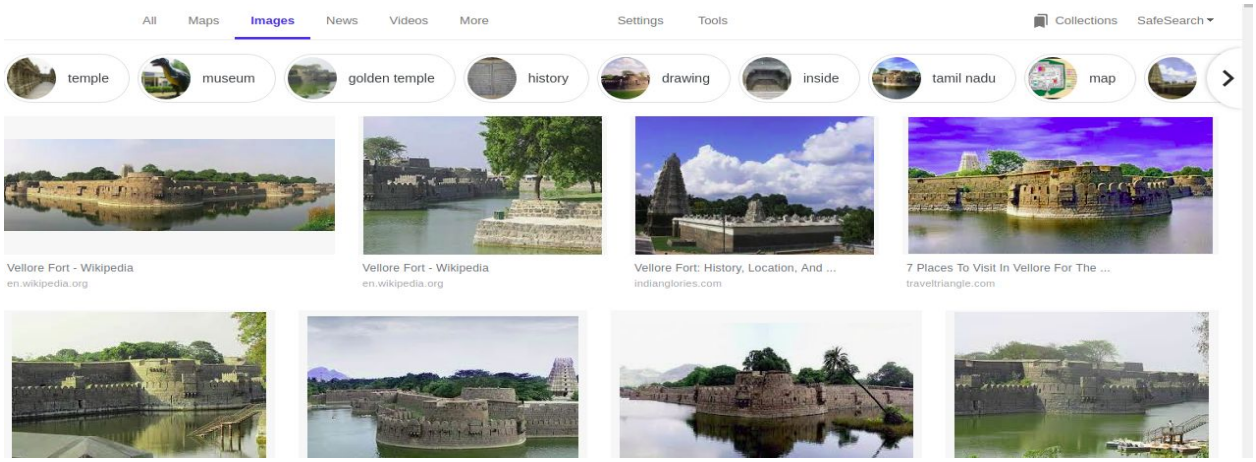
What is Linked Data and the Semantic Web and what is it all about?

To a computer, the Web is a flat, boring world, devoid of meaning. As in fact documents on the Web describe real objects and imaginary concepts. Adding semantics to the Web involves two things: allowing documents which have information in machine-readable form, and allowing links to be created with relationship values. Only when we have this extra level of semantics, we will be able to use computer power to help us exploit the information to a greater extent than our own reading.

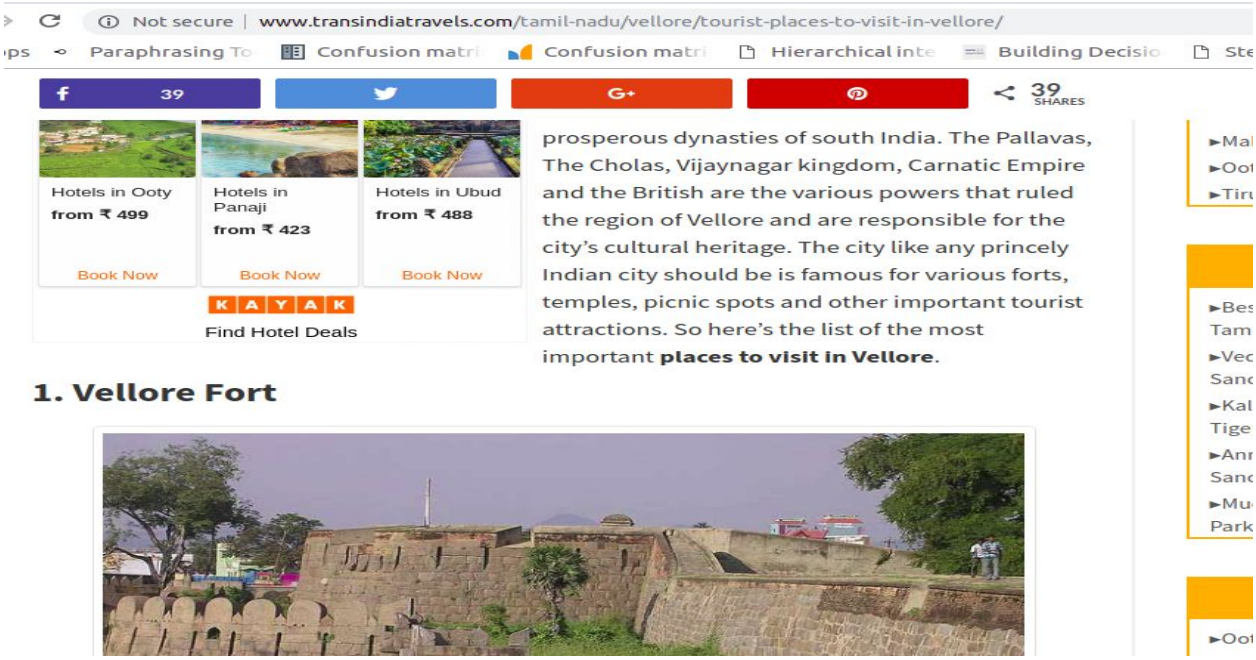
Now let us take an example of searching for photographs of some tourism place, we can search in Flickr,



Google



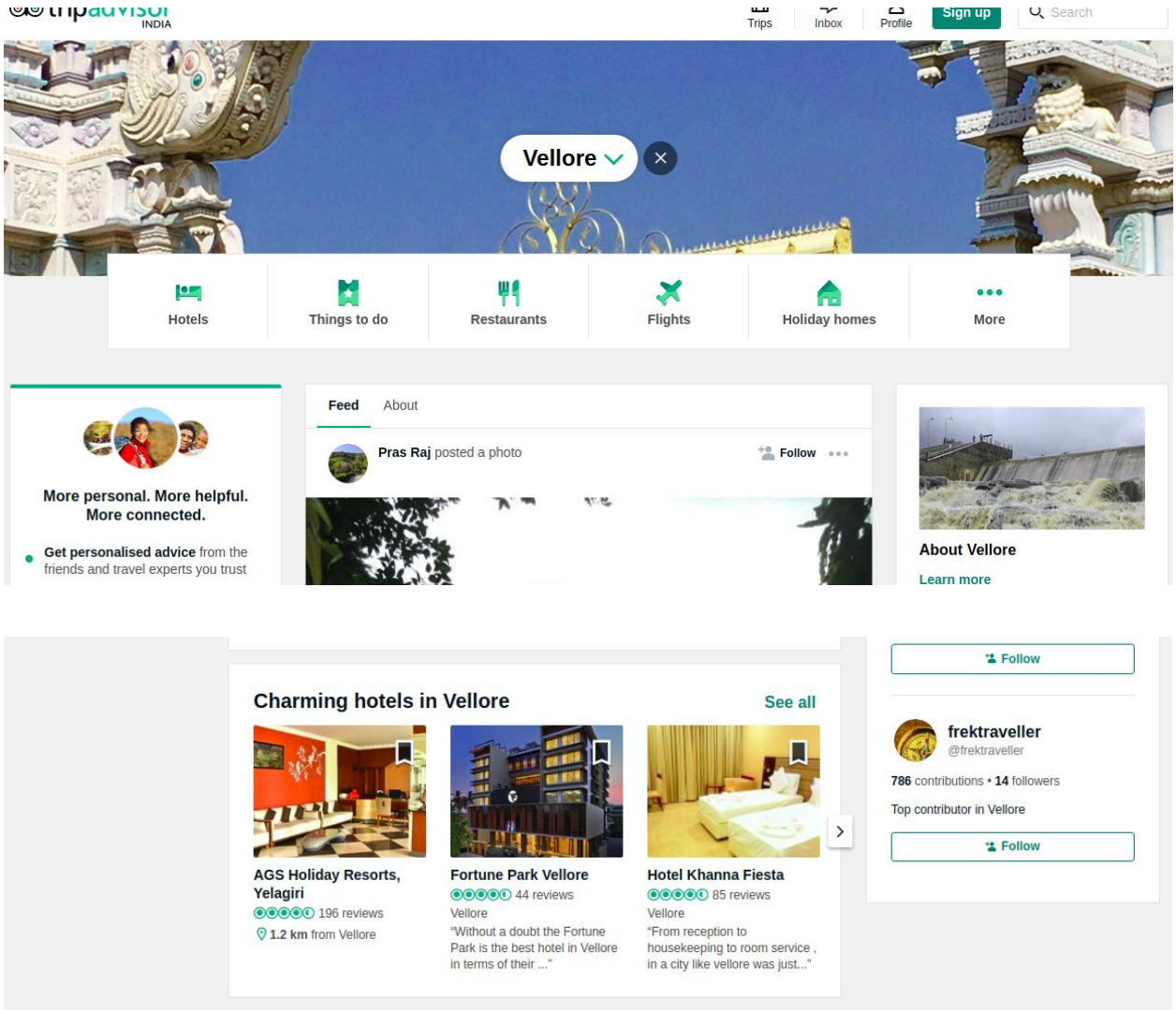
and maybe in any social travel sites.



What happens here is that we had to consult a large number of sites, all different in style, purpose, possibly language and mentally integrate all those information to achieve your goals. This is a long and tedious process!

All those pages are only the tip of respective icebergs. The real data is hidden somewhere in databases, XML files, Excel sheets, etc. we have only access to what the Web page designers allows us to see.

There are some specialized websites like Expedia, TripAdvisor does a bit more, they gather and combine data from other sources (usually with the approval of the data owners)



but they still control how you see those sources. But sometimes you want to personalize: access the
original data and combine it yourself! **Therefore the data has to be "published" or able to "publish" data in a common standardized way instead of an ad-hoc way.**

People create different documents they give an address to it (ie, a URI) and make it accessible to others on the Web.

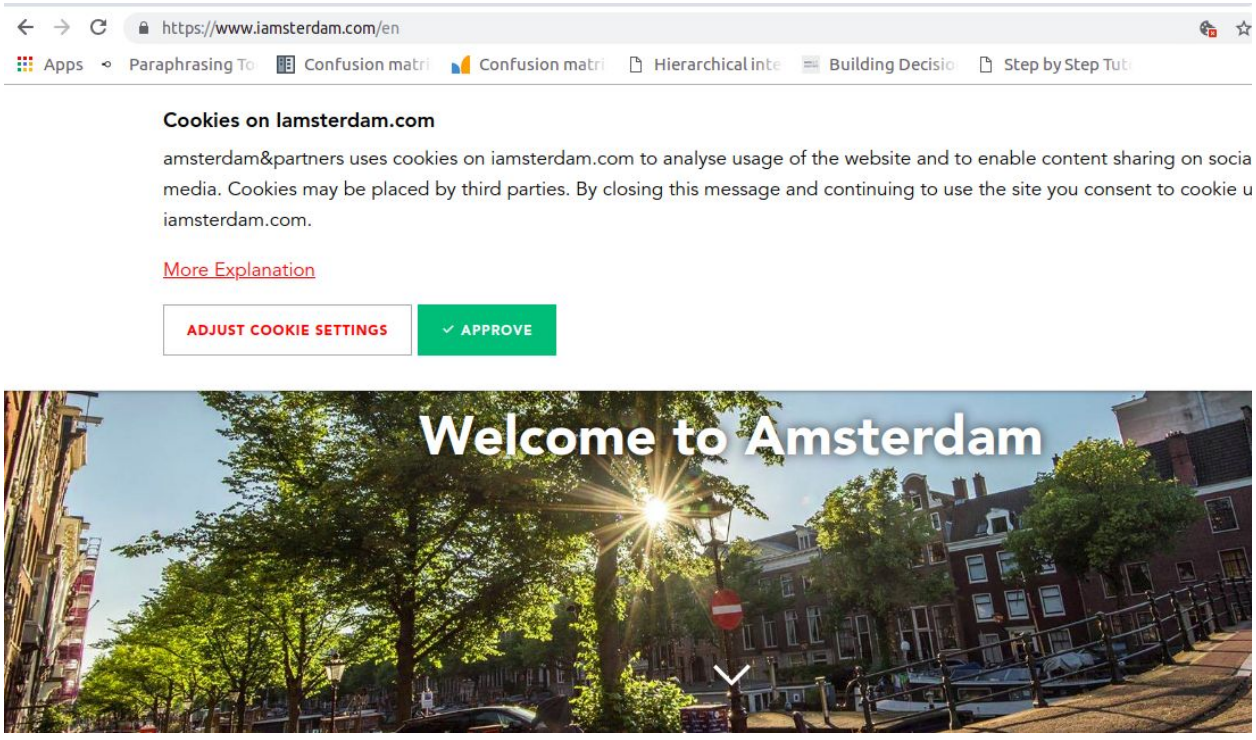
Now a question might arise why URI is given more importance in publishing data onto a web?

The fundamental value and differentiating capability of the Semantic Web is the ability to connect things. The URI is what makes this possible.

> A Uniform Resource Identifier (URI) is important to the semantic web because it prevents ambiguity. By using a URI, we know for certain that we're talking about the same thing.
>
> URIs can be both Uniform Resource Locators (URLs) and Uniform Resource Names (URNs). The difference is that a URL tells you how to access the resource, while a URN does not. You can think of a URN as the unique name of a thing, and the URL as the address.
>
> A web link, such as 'http://example.com/' is a URL (and a URI): it tells you that you can access the resource by using the HTTP protocol.
>
> An ISBN number, such as 'ISBN: 026201210' is a URN (and a URI): although it is an identifier, it does not tell you how you can find it.

Once we are able to publish the data, we should be able to "link" the related resources, in order to achieve interoperability.

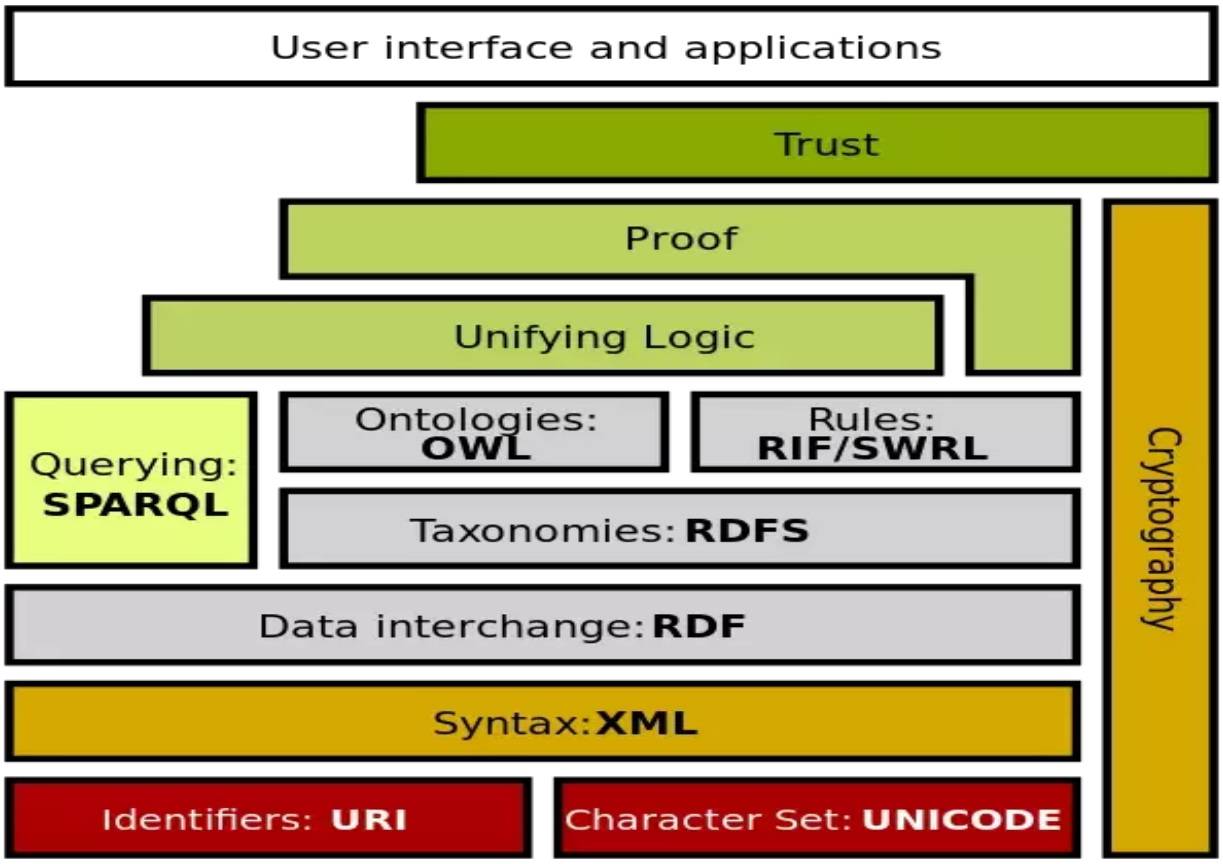
There is something called as "context" that each person may use. For example, I have an address text as links to another website of my residential place



information here the “context” is to link each address to its residential place information but the machine can’t understand. **Therefore, we need to characterize both link and the target with some “label” or classify the reasons for linking.**

By these we can extend the current Web to a standard way for a “Web of data” and Web Semantics is a collection of standard technologies to realize a Web of Data.

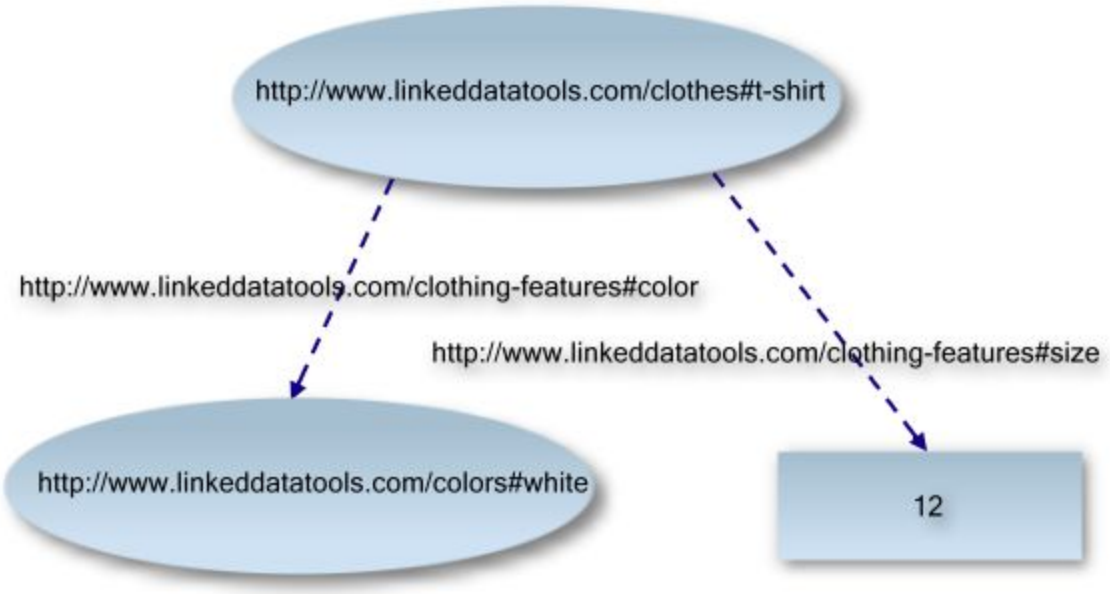
The basic architecture of web semantics:



From the components, there are 4 main components we might wanna look into deep for building an ontology or any knowledge graph base.

Data interchange: RDF [Resource Description Framework]
a general method for describing information. RDF is a simple language for expressing data models, which refer to objects ("web resources") and their relationships. An RDF-based model can be represented in a variety of syntaxes, e.g., RDF/XML, N3, Turtle, and RDFa. RDF is a fundamental standard of the Semantic Web.

RDF is the data model of the Semantic Web. It means that all data in Semantic Web technologies are represented as RDF. If we store Semantic Web data, it's in RDF. If we query Semantic Web data (typically using SPARQL), it's RDF data. If you send Semantic Web data to your friend, it's RDF




```
<rdf:RDF
    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
    xmlns:feature="http://www.linkeddatatools.com/clothing-features#">

    <rdf:Description rdf:about="http://www.linkeddatatools.com/clothes#t-shirt">

        <feature:size>12</feature:size>
        <feature:color
        rdf:resource="http://www.linkeddatatools.com/colors#white"/>

    </rdf:Description>

</rdf:RDF>
```

Taxonomies: RDFS [RDF Schema]

RDF Schema extends RDF and is a vocabulary for describing properties and classes of RDF-based resources, with semantics for generalized-hierarchies of such properties and classes.

From the previous example, we can specify something like this with owl where shirl is a subclass of men’s fashion.

```
<rdf:Description rdf:ID="men's fashion">
  <rdf:type
   rdf:resource="http://www.w3.org/2000/01/rdf-schema#Class"/>
</rdf:Description>

<rdf:RDF
        xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
        xmlns:feature="http://www.linkeddatatools.com/clothing-features#">

        <rdf:Description rdf:about="http://www.linkeddatatools.com/clothes#t-shirt">
                <rdfs:subClassOf rdf:resource="#men's fasion"/>

                <feature:size>12</feature:size>
                <feature:color rdf:resource="http://www.linkeddatatools.com/colors#white"/>

        </rdf:Description>

</rdf:RDF>
```

Ontologies: OWL [Web Ontology Language]

A family of knowledge representation languages. OWL adds more vocabulary for describing properties and classes: among others, relations between classes (e.g. disjointness), cardinality (e.g. "exactly one"), equality, richer typing of properties, characteristics of properties (e.g. symmetry), and enumerated classes.

An ontology might include a rule, for example, expressing that no person can be the mother of another person without also being a parent of that person. The hope was that these ontologies would be widely used not only to check the accuracy of RDF found in the wild but also to make inferences about omitted information

Now we can see an owl in ttl format for the statement: "Pizza has PizzaBase as its base; Pizza is disjoint with PizzaBase; NonVegetarianPizza is exactly Pizza that is not VegetarianPizza; isIngredientOf is a transitive property; isIngredientOf is inverse of hasIngredient".

```
@prefix :     <http://example.com/pizzas.owl#> .
@prefix rdf:  <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix owl:  <http://www.w3.org/2002/07/owl#> .

:Pizza rdfs:subClassOf
        [ a owl:Restriction ;
          owl:onProperty :hasBase ;
          owl:someValuesFrom :PizzaBase ] ;
       owl:disjointWith :PizzaBase .

:NonVegetarianPizza  owl:equivalentClass
        [  owl:intersectionOf
            (  [owl:complementOf :VegetarianPizza]
               :Pizza  )  ] .

:isIngredientOf
      a owl:TransitiveProperty , owl:ObjectProperty ;
      owl:inverseOf :hasIngredient .
```

Querying : SPARQL
An RDF query language

Example to query all the triplets from the t-shirt example above:

```
PREFIX:http://www.linkeddatatools.com/clothing-features#
SELECT DISTINCT
?s ?p ?o
WHERE {
?s ?p ?o .
}
```

Where do the web crawlers fail?

Web crawlers are computer programs that scan the web, ‘reading’ everything they find. But not all the data there are challenges like:

- Data formats and structures are inconsistent in the ever-evolving Web space [Non -uniform structures].
- Difficulties in crawling complex pages with dynamic content. Think about on-site search results, Flash content, forms, animations, and other dynamic resources.

Why does semantic web/graph be not so a popular technology even though it was started a quite long back?

Let us see some common reasons why semantic graph/web is failing.

- **The languages for encoding metadata were too complex. Moreover, encoding metadata was time-consuming and prone to errors.** The proposed languages for adding metadata to web pages and resources were difficult to use. Despite the availability of some authoring tools, describing knowledge was not easy.

- **Languages were very restrictive**. Other problems are that since some of these languages were based on logic, they were very restrictive. To describe some very simple knowledge it would work fine. But to describe something complicated, it was actually very hard to model something properly.
  .
- From the industry point of view, the value of the semantic web is only when content owners open up data to others so that it can be connected with other datasets. But to get some any internal knowledge why would an enterprise need to publish their information. To prevent publishing sensitive information there are ways to hide our information inside our RDF through a concept called Blank Nodes but it might cause a few more complexities while merging graphs or querying.

Apart from these disadvantages, there is some useful or interesting information in the semantic web which can be fine-tuned and used at an enterprise level. This is where the concept of Labelled Property Graph [LPG] arises, which is nothing but a graph with vertices and edges which uses RDF in a quite lesser complex way.