

Practical session (1): RDF

Sources: <http://www.linkeddatatools.com/introducing-rdf>

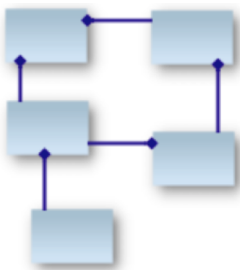
1. Introduction

RDF is a common acronym within the semantic web community because it forms one of the basic building blocks for forming the web of semantic data. What it defines is a type of database which you may not be immediately familiar with: something called a **graph database**.

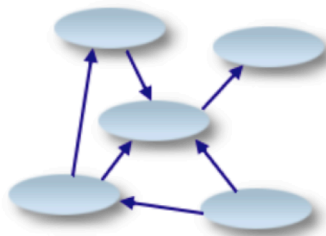
Although it might not be familiar to you, it is the type of database that builds the semantic web, globally. We will learn why in these tutorials.

This document will help you to learn what a graph database is, how RDF defines one, and visualise graph data so you can get a feel of what it looks like. We will begin by comparing hierarchical, relational, and graph databases to see how they are different.

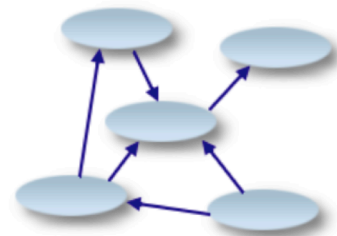
1.1 Introducing The Graph Knowledge Bases



Relational DB:
Tables related by
foreign keys



Hierarchical KB:
Parent Nodes have
more intrinsic importance



Graphs KB:
Resources/objects related
by object properties

For most types of data storage, there is the concept of some elements of data (whether they be for example data nodes or data tables) having more precedence, or importance, over other elements.

For example, take an XML document. An XML document typically contains nodes of information each with a parent node. At the root of the document is the highest level node, which has no parent.

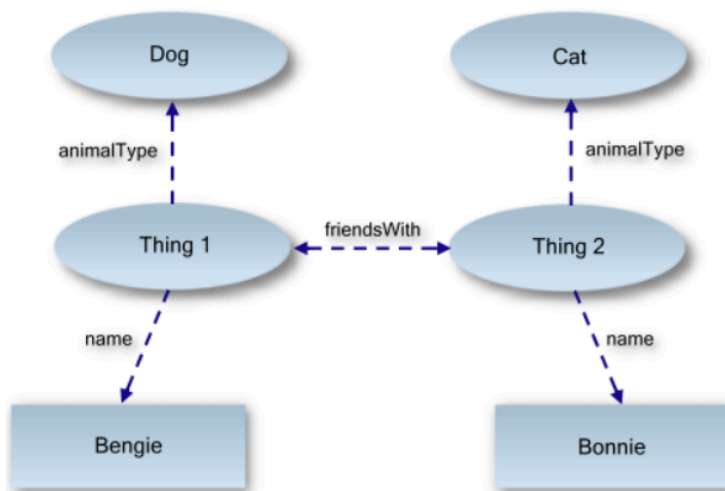
In a data graph, there is no concept of roots (or a hierarchy). A graph consists of resources related to other resources, with no single resource having any particular intrinsic importance over another.

Example of a Data Graph

Look at the following statements describing the relationship between a dog (called Bengie) and a cat (called Bonnie):

Bengie is a dog.
Bonnie is a cat.
Bengie and Bonnie are friends.

Using these three simple statements, let's turn this into a data graph:



We can see that our two things - identified by "Thing 1" and "Thing 2" - have the properties **name**, **animalType** and **friendsWith**.

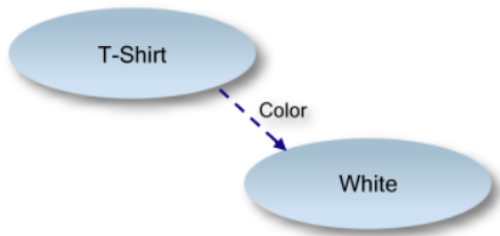
```
<?xml version="1.0" encoding="UTF-8"?>
<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:dc="http://purl.org/dc/elements/1.1/"
  xmlns:region="http://www.country-regions.fake/">
  <rdf:Description rdf:about="http://en.wikipedia.org/wiki/Oxford">
    <dc:title>Oxford</dc:title>
    <dc:coverage>Oxfordshire</dc:coverage>
    <dc:publisher>Wikipedia</dc:publisher>
    <region:population>10000</region:population>
    <region:principaltown rdf:resource="http://www.country-regions.fake/oxford"/>
  </rdf:Description>
</rdf:RDF>
```

The RDF Statement (Triple)

The RDF/XML above (between the `<rdf:Description>` tags) is called an RDF statement, or sometimes called an RDF triple. Of the two, triple is the most helpful term as it describes the breaking of the

statement into its three constituent parts: the subject, predicate, and object of the statement.

It is easiest first to illustrate these terms in the form of a simple graph. Look at the following graph of data describing the color of a T-shirt:



In terms of the simple graph above, the:

- **Subject** is the T-shirt
- **Predicate** (property) is the color
- **Object** is white

```
<?xml version="1.0" encoding="UTF-8"?>
<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:color rdf:resource="http://www.linkeddatatools.com/colors#white"/>
  </rdf:Description>
</rdf:RDF>
```

2. Building an RDF Document

- Add The RDF Document Root Tag :

```
<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  <!-- Body Code Omitted -->
</rdf:RDF>
```

If you look at line 02, you will see the standard W3.org namespace <http://www.w3.org/1999/02/22-rdf-syntax-ns/>. This namespace tells any machine reader that the enclosing document is an RDF document, and that the `rdf:RDF` tag resides in this namespace. This namespace, and the RDF node, forms the root of all RDF documents.

- Add A Statement

An RDF document can contain more than one statement. For simplicity, we'll only add one. Start by adding a an `rdf:Description` tag, which in RDF/XML can contain one or more statements about the same subject:

```
<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  <rdf:Description rdf:about="http://www.linkeddatatools.com/clothes#t-shirt">
    <!-- Statement Code Omitted -->
  </rdf:Description>
</rdf:RDF>
```

The `rdf:Description` tag simply means "I'm going to describe something (a subject) and I'm giving it the unique ID <http://www.linkeddatatools.com/clothes#t-shirt>".

```
<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>
  </rdf:Description>
</rdf:RDF>
```

See line 07. This simply says "The subject has a property with name `feature:size` which has the literal value 12". In RDF terminology, this is a statement.

Finally, let's add one more predicate: the color of the T-shirt.

Finally, let's add one more predicate: the color of the T-shirt.

```
<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>
```

Breaking Down The Statement

Now we've looked at a simple example of an RDF document, let's formalize what we've learned and break the statement into its component parts:

```
<rdf:Description rdf:about="subject">
  <predicate rdf:resource="object" />
  <predicate>literal value</predicate>
</rdf:Description>
```

Note: the `rdf:Description` RDF/XML element allows you to group one or more statements into a single container. The above general form actually contains two statements referring to the same subject, but with two predicates and objects: a resource and a literal.

A More Thorough Example

To test your understanding and point out any areas where you need to recap, see if you can identify on the RDF document above the:

- Subject of the statement
- Predicates of the statement - including whether they are resources or literals
- Objects referenced by the resource predicates

- Once you have understood RDF documents, and how they relate to data graphs, you are ready for our next lesson introducing the idea of modelling semantics into RDF graph data.

```
<?xml version="1.0" encoding="UTF-8"?>

<rdf:RDF

  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"

  xmlns:dc="http://purl.org/dc/elements/1.1/"

  xmlns:region="http://www.country-regions.fake/">

  <rdf:Description rdf:about="http://en.wikipedia.org/wiki/Oxford">

    <dc:title>Oxford</dc:title>

    <dc:coverage>Oxfordshire</dc:coverage>

    <dc:publisher>Wikipedia</dc:publisher>

    <region:population>10000</region:population>

    <region:principaltown rdf:resource="http://www.country-regions.fake/oxford"/>

  </rdf:Description>

</rdf:RDF>
```

A Quick Recap Of URIs And XML Namespaces

Unique IDs that we've been using so far such as <http://www.linkeddatatools.com/clothes#t-shirt> are called Uniform Resource Identifiers, or URIs for short. We've been using URIs to give a unique ID to the subjects, predicates or objects of statements so far without really saying why.

Because URIs are so crucial to the driving purpose behind RDF - to make data exchangeable globally - we will make a quick recap of URIs now. If you already understand URIs, you can skip this section.

XML Namespace URIs

Look back again at our example RDF document we've built. See that the T-shirt size predicate has the name [feature:size](#) on line 07 below:

```
<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>
```