RDF / TURTLE

What is RDF ?

What even mean ?

Resource Description Framework

and Turtles ?

Terse RDF Triple Language

Syntax for the RDF Model!

RDF

what is the purpose of it ?

distributed information

distributed information

semantic web

distributed information

semantic web

graph representation

distributed information

semantic web

graph representation

fragmentation of information

distributed information

semantic web

graph representation

fragmentation of information

mechanic communication

who is constructed ?

Triples of Knowledge

List of Tuples with 3 Elements

SUBJECT PREDICATE OBJECT

[(SUBJ PRED OBJ)]

SUBJECT PREDICATE OBJECT

structure

Express information as a list of statements

statements

SUBJECT PREDICATE OBJECT

I own my_apartment

my_apartment has my_computer

my_apartment has my_bed

my_apartment is_in Philadelphia

I own my_apartment

my_apartment has my_computer

my_apartment has my_bed

my_apartment is_in Philadelphia

as a graph like structure

SUBJECT PREDICATE OBJECT

I -> own -> my_apartment

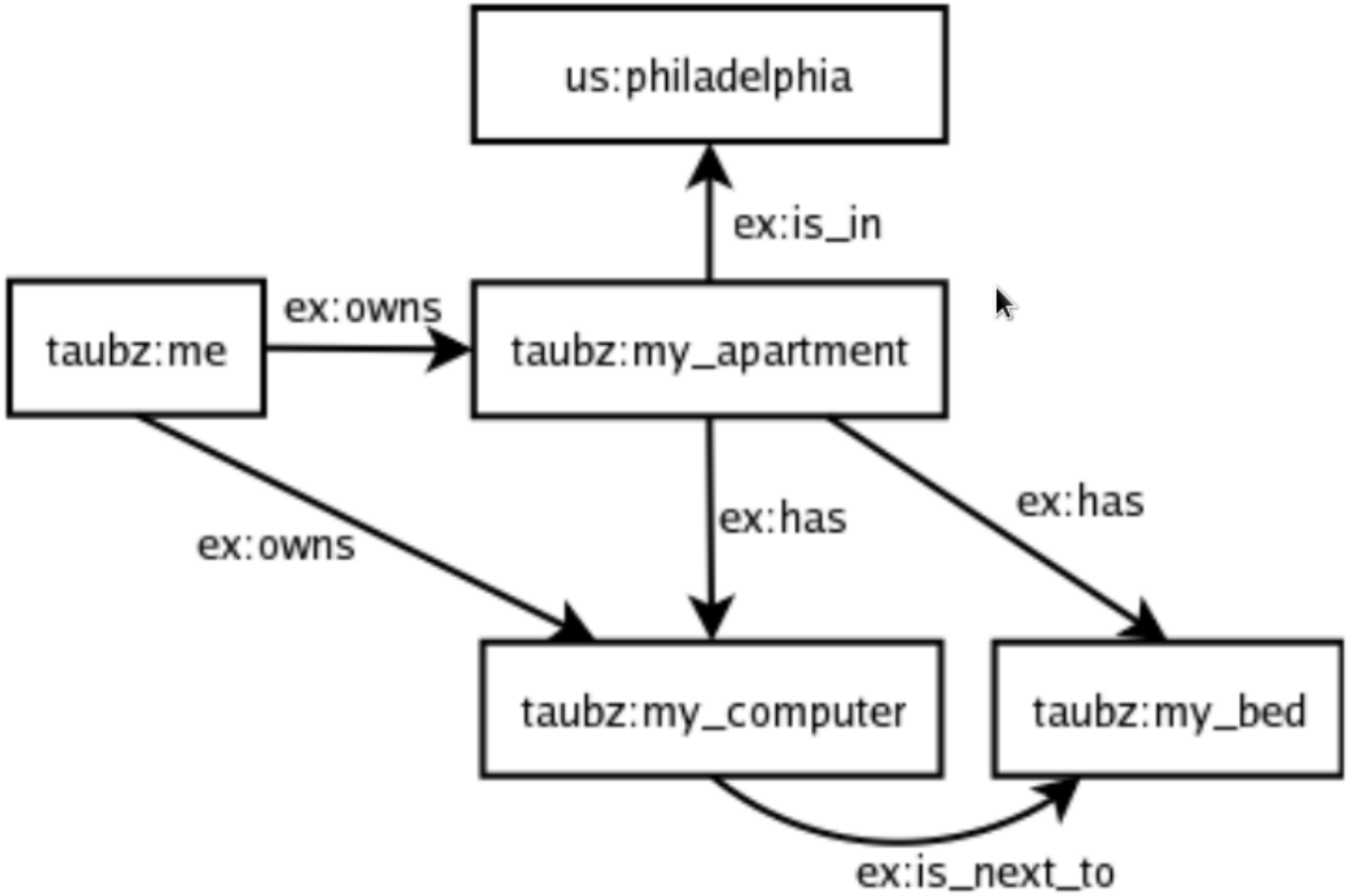
my_apart -> has -> my_computer

my_apart -> has -> my_bed

my_apart -> is_in -> Philadel

global scope of eatch

Subject and Object


us:philadelphia
ex:is_in
taubz:me
ex:owns
taubz:my_apartment
ex:owns
ex:has
ex:has
taubz:my_computer
taubz:my_bed
ex:is_next_to

what is this good for ?

semantic and systematic search

semantic meaning

like English words or phrases

```
quest = (my_apart, has, what)

for each stat in kldg {

 if ((stat.subject == quest.subject)

... ) {

   call FoundAwnser(stat)

   }
```

Output:

my_apartment has my_computer

my_apartment has my_bed

made to work on web

URIs

URIs as Subjects

URIs as Objects

example of

http://example.org/has

what so far

.1 A fact is expressed as a triple of the form

(Subject, Predicate, Object).

.2 Subjects, predicates, and objects are given as names for entities, whether, concrete or abstract, in the real world.

.3 Names are in the format of URIs, which are opaque and global.

values of objects can been

Literals

Literals can ben wharever raw text

or value you want to

we can use black nodes or

better to say,

a local value that only exist on this tuple, but is not global

_:anon123 foaf:name "some one"

_:anon123 ex:has_read "books"

_:anon123 dc:autor _:anon321

_:anon321 foaf:name "Sr Carol"

this is like local values

we can use quotation mark for literals

anonymous nodes with ’_’

literals as raw textual representations

untyped and typed literals for parsing and context

with URI

NTriples

With N-Triples

we can abreviate stuff

prefix

@prefix

@prefix rdf:

<http://www.w3.org/1999/02/22-rdf-syntax-ns#>

use comas as a list kind data

http://br.org tafl:has "32", "42"

or use brackets

```
@prefix br http://br.org

br [ tafl:has "32", "42" ]
```