# INDEXING AND SEARCHING RDF DATASETS

## Improving Performance of Semantic Web Applications with Lucene, SIREn and RDF

Mike Hugo
Entagen, LLC

ENTAGEN
ACCELERATING INSIGHT

slides and sample code can be found at

https://github.com/mjhugo/rdf-lucene-siren-presentation

ENTAGEN
ACCELERATING INSIGHT


Groovy


HIBERNATE


Java
Sun Microsystems


JBoss™


ORACLE®


GRAILS

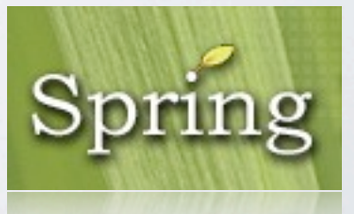

MySQL®


Apache
Solr


Lucene


N
W
E
S



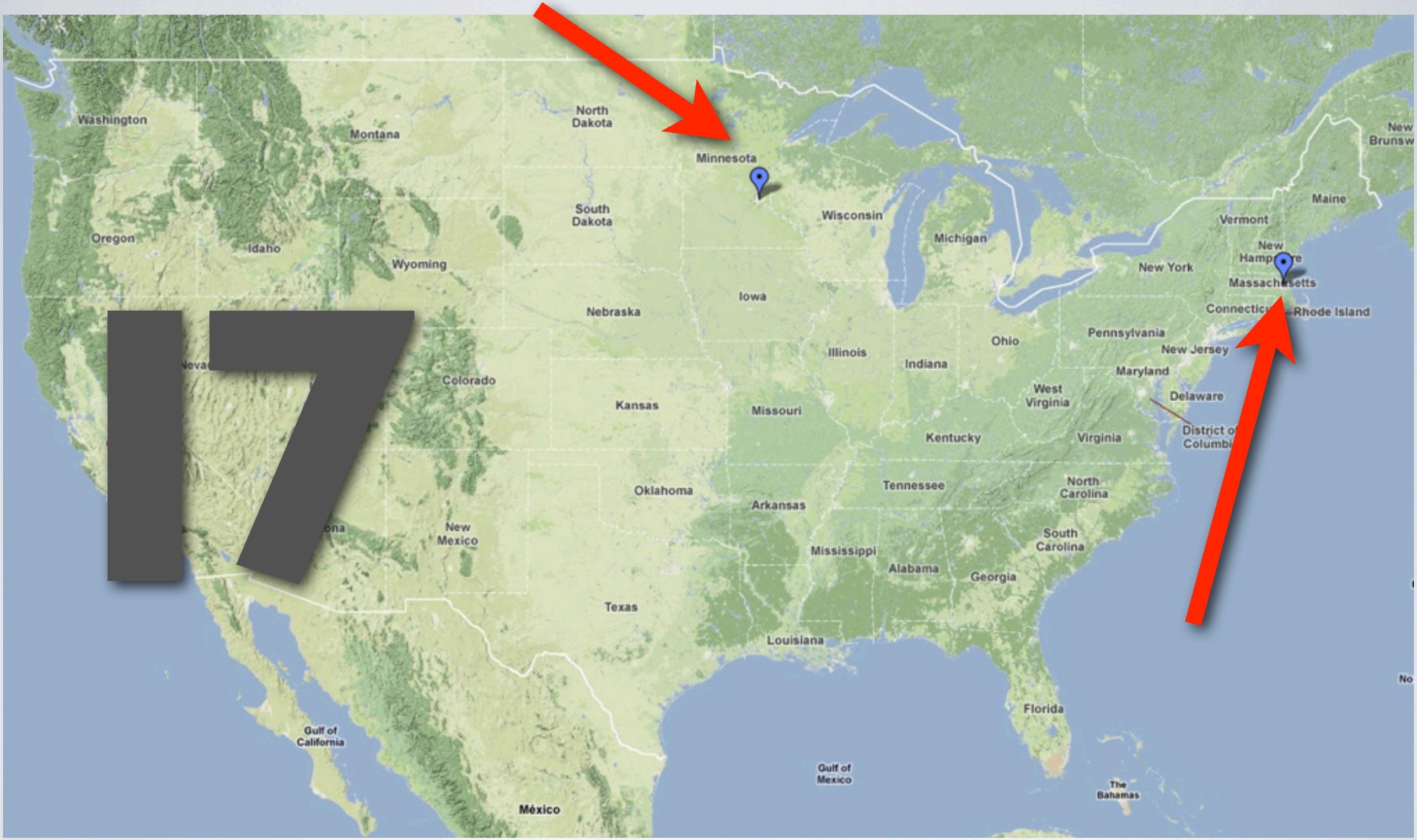

Spring


ENTAGEN

# ENTAGEN

ACCELERATING INSIGHT

ENTAGEN
ACCELERATING INSIGHT




ENTAGEN
ACCELERATING INSIGHT

MERCK

VULCAN

LabCorp
Laboratory Corporation of America

NOVARTIS

Boehringer Ingelheim

AILERON THERAPEUTICS

Pfizer

Cequent pharmaceuticals

MILLIPORE

Dart NeuroScience LLC
quality of mind

MERCK

DANA-FARBER CANCER INSTITUTE

ENTAGEN
ACCELERATING INSIGHT

# TripleMap

Connections Matter

ENTAGEN
ACCELERATING INSIGHT

# AGENDA

ENTAGEN
ACCELERATING INSIGHT

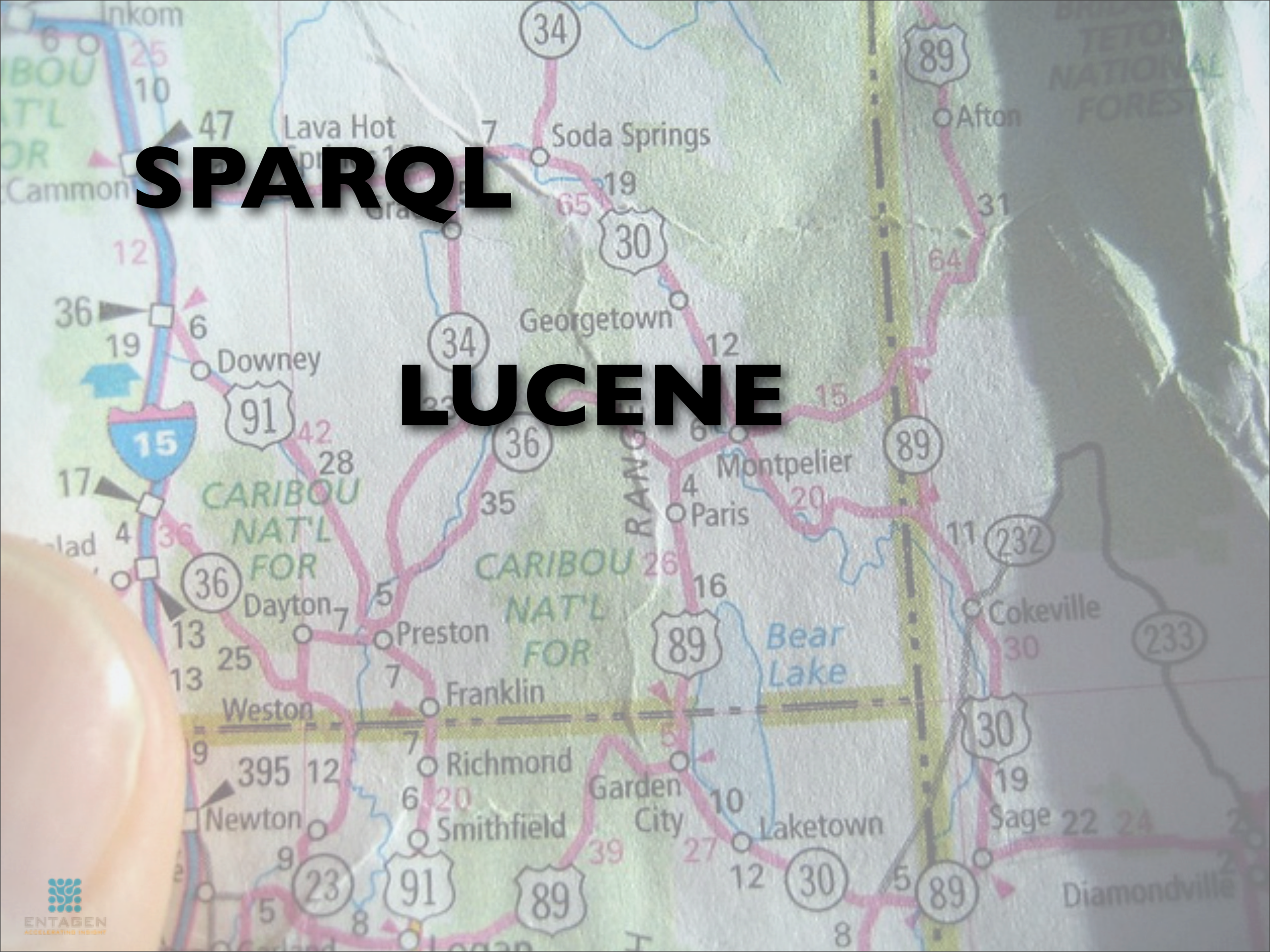
Inkom
25
10
47
Lava Hot Springs
16
11
7
Soda Springs
19
Cammon
Grace
5
65
34
89
Afton
31
64
30
12
36
19
6
Downey
91
Georgetown
34
12
33
15
42
28
15
6
Montpelier
89
17
4
CARIBOU NAT'L FOR
35
RANGE
4
Paris
29
36
CARIBOU NAT'L FOR
26
16
11
232
Cokeville
233
30
Dayton
5
7
Preston
89
Bear Lake
13
25
13
7
Franklin
Weston
30
9
7
5
19
395
12
Richmond
Garden City
10
Laketown
Sage
22
24
Newton
6
20
Smithfield
39
27
9
23
91
89
12
30
5
89
Diamondville
8
5
8

# SPARQL


Inkom
34
89
Afton
Lava Hot
Soda Springs
Cammon
19
65
30
31
64
12
36
19
6
Downey
91
34
Georgetown
12
33
15
15
42
28
CARIBOU
NAT'L
FOR
36
89
Montpelier
17
4
Paris
20
11
232
Dayton
Preston
Cokeville
13
25
13
7
5
7
89
Bear
Lake
233
30
Weston
Franklin
9
7
Richmond
395
12
6
20
Garden
City
10
Laketown
19
Sage
22
Newton
Smithfield
9
23
91
89
39
27
12
30
89
Diamondville
8
8

# SPARQL

# LUCENE




SPARQL
LUCENE
SIREN
Inkom
Lava Hot
Soda Springs
Afton
Georgetown
Downey
Montpelier
Paris
Dayton
Preston
Franklin
Weston
Richmond
Garden City
Laketown
Newton
Smithfield
Cokeville
Sage
Diamondville
CARIBOU NAT'L FOR
ENTAGEN
ACCELERATING INSIGHT

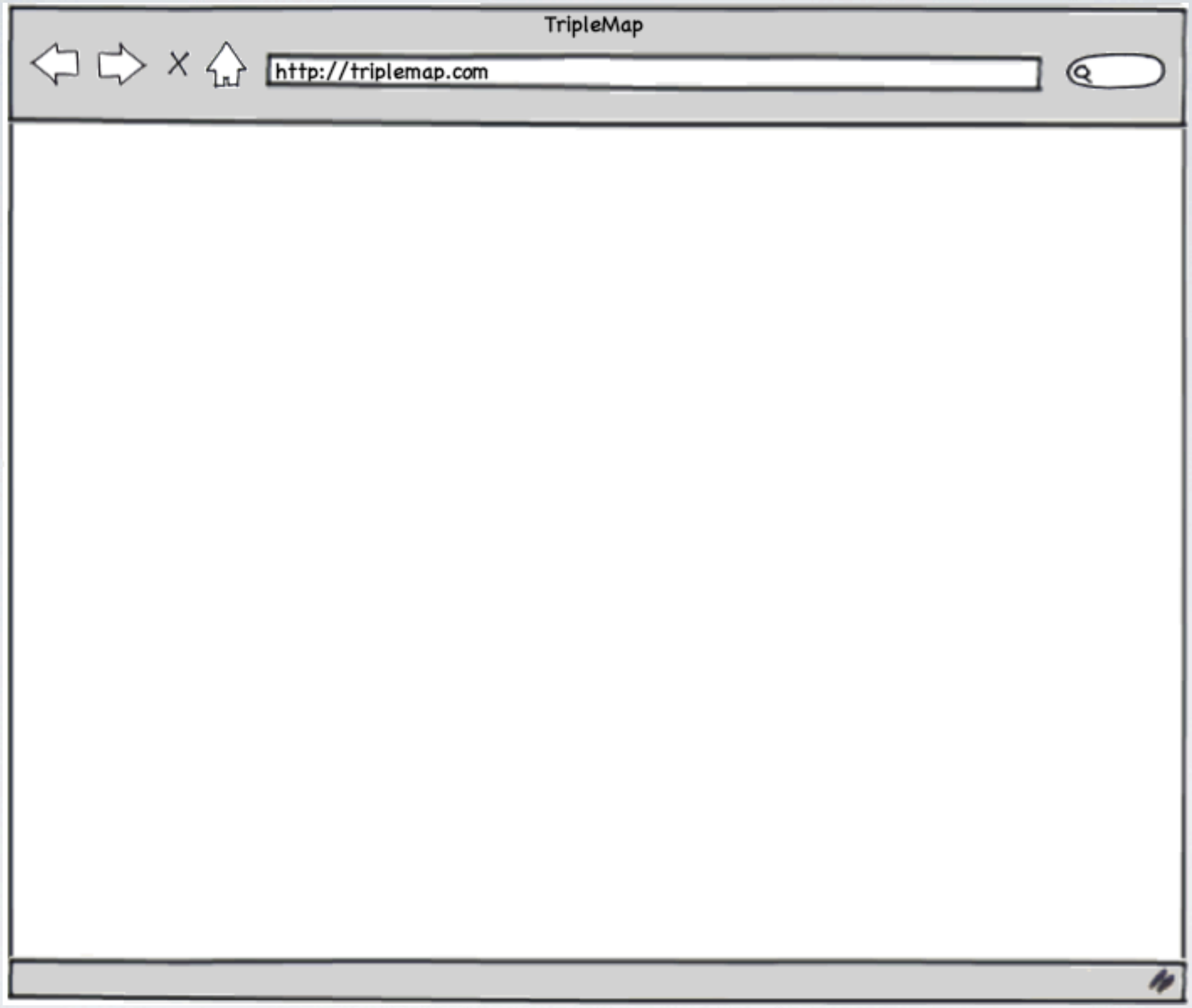
# SPARQL

# LUCENE

# SIREN

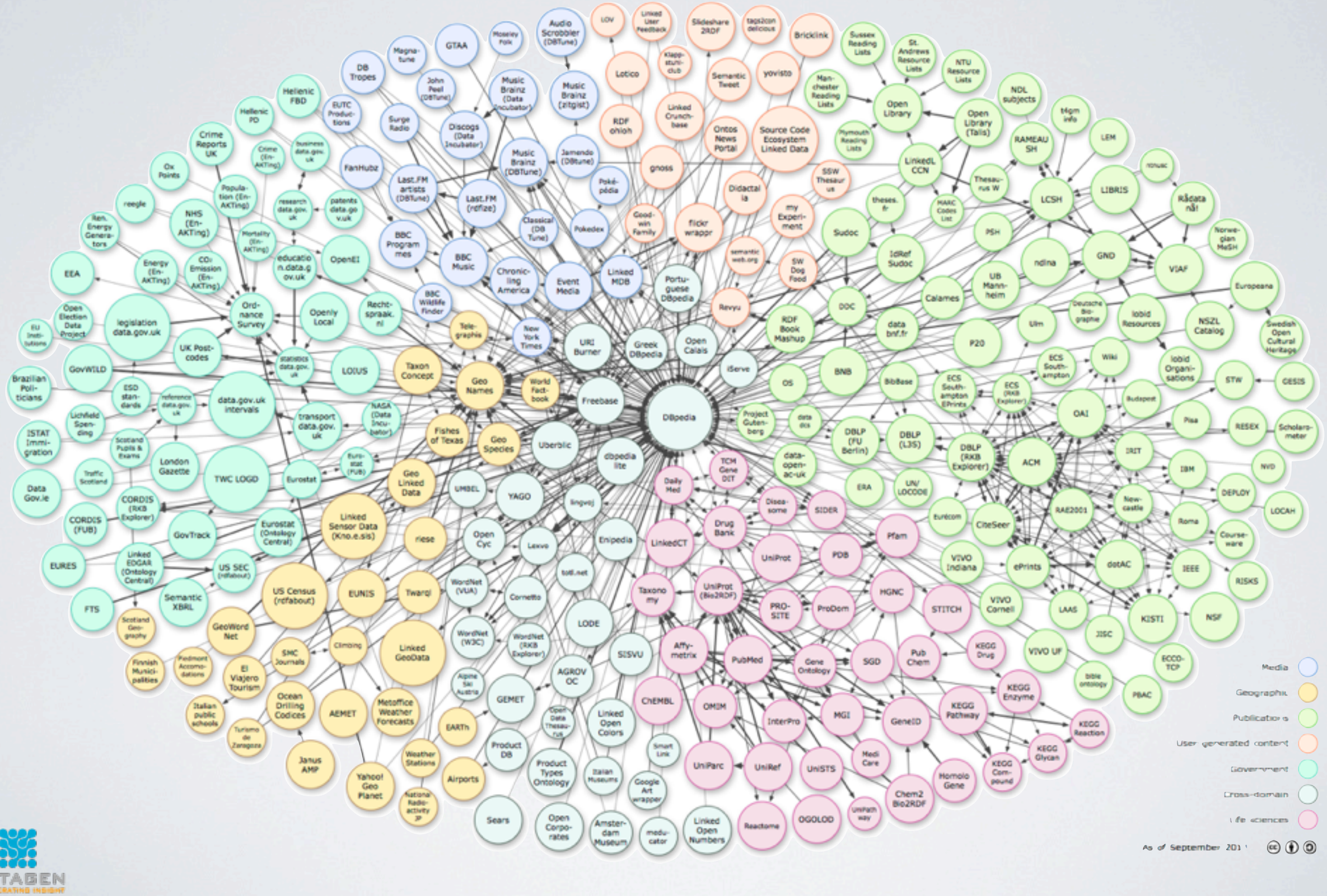
# TripleMap.com

ENTAGEN
ACCELERATING INSIGHT



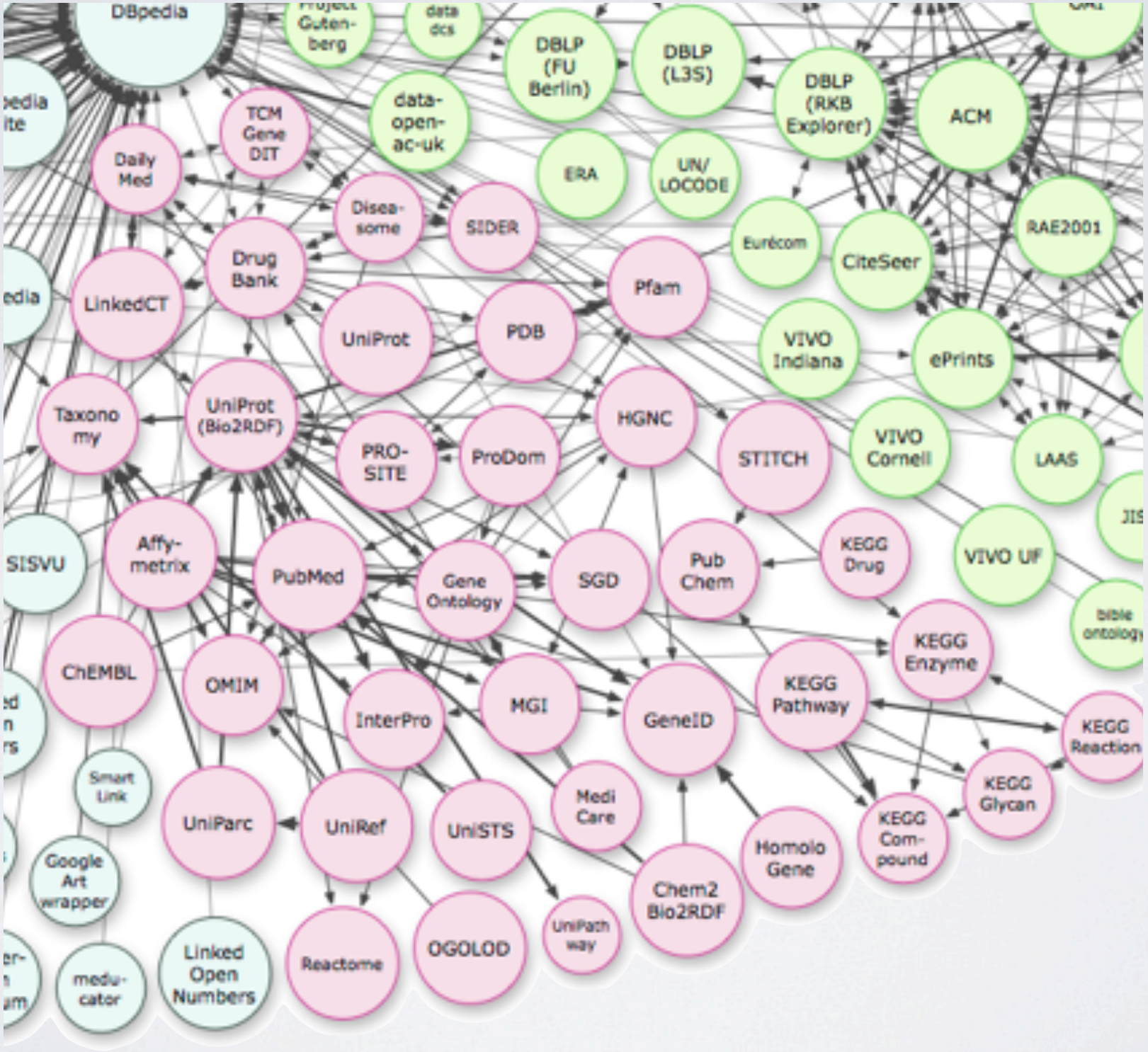

ENTAGEN
ACCELERATING INSIGHT

# LINKING OPEN DATA



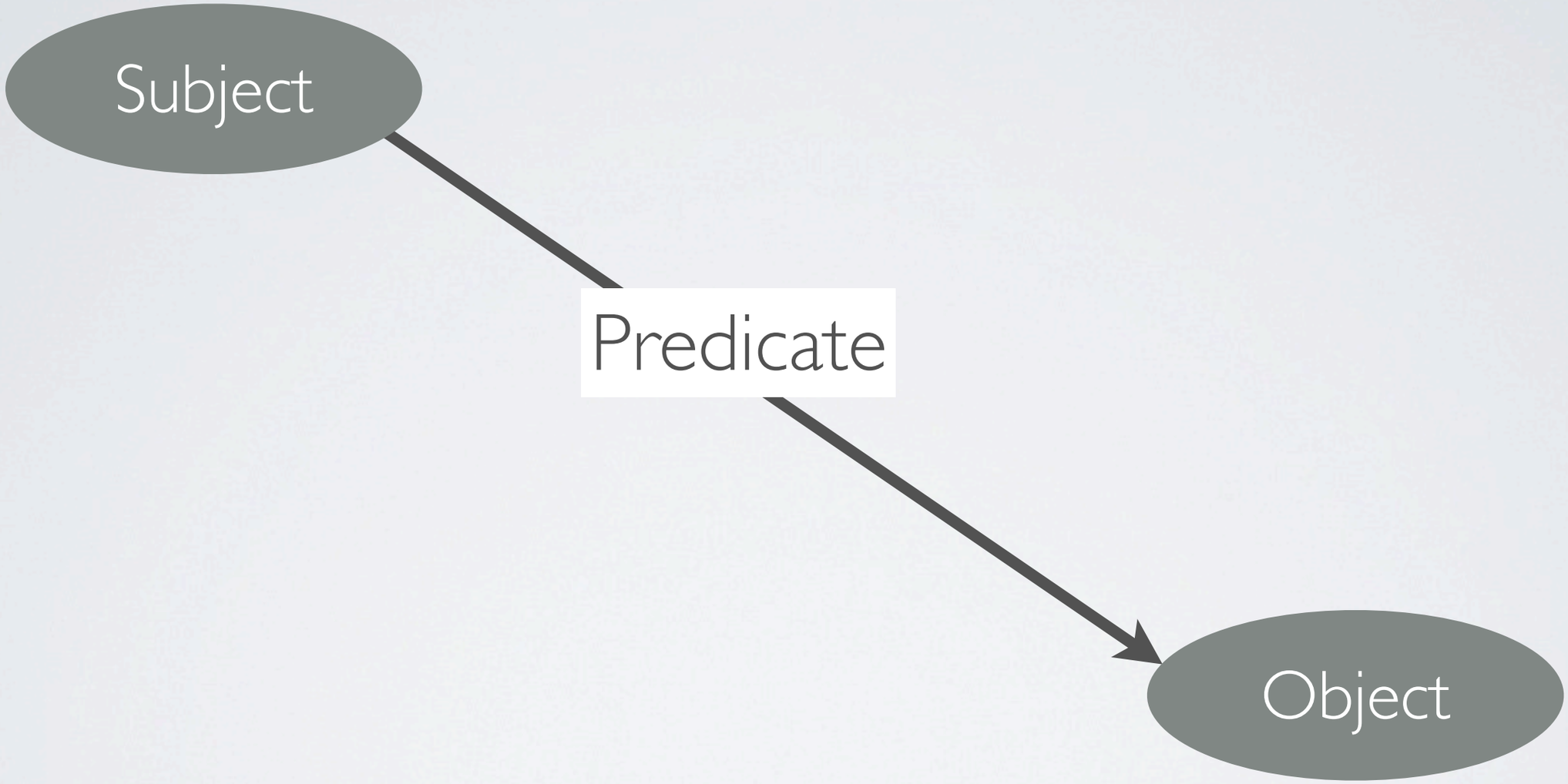

ENTAGEN
ACCELERATING INSIGHT

# **LIFE SCIENCE** LINKED DATA



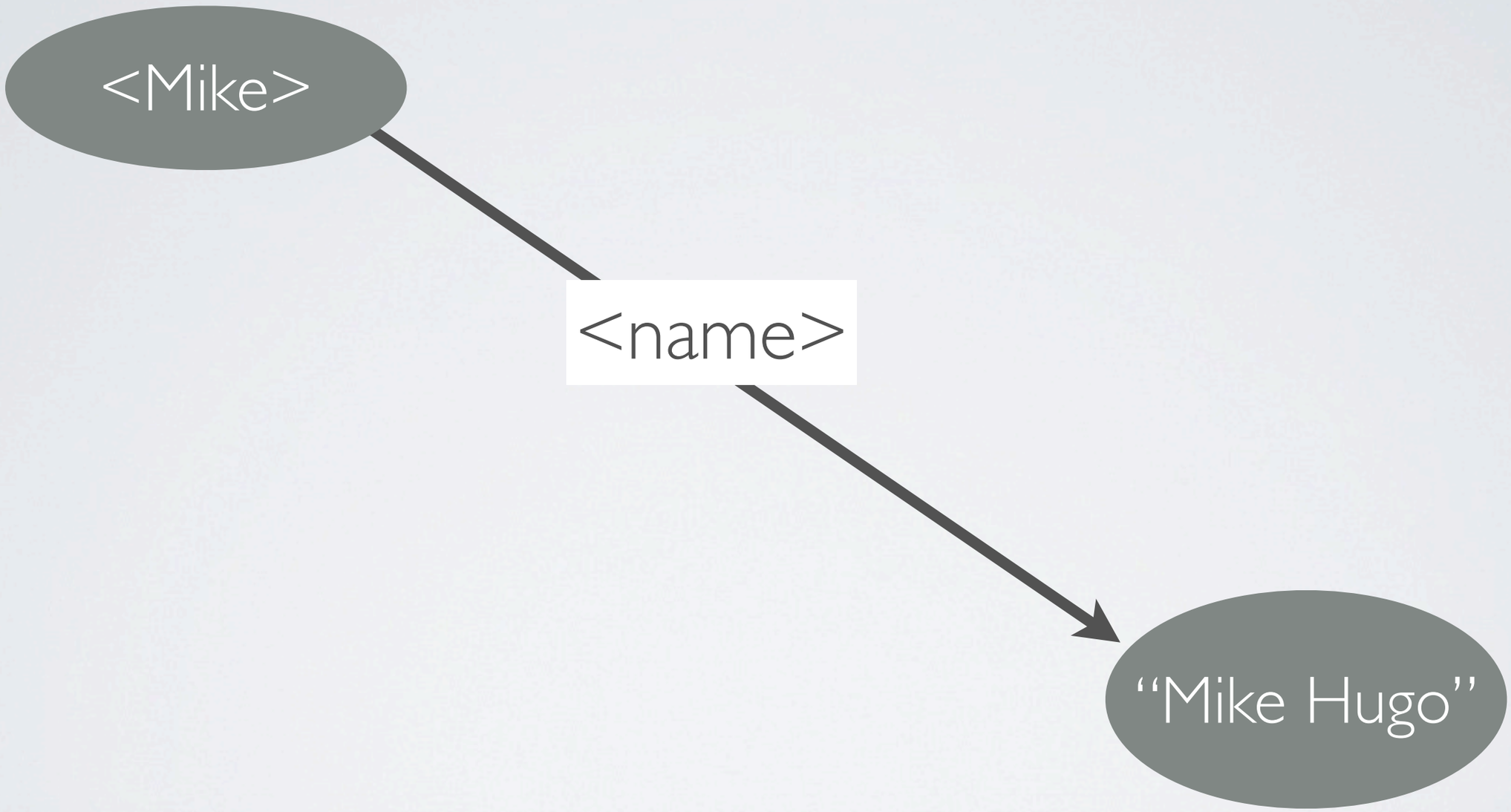

ENTAGEN
ACCELERATING INSIGHT

# WHAT'S A TRIPLE?

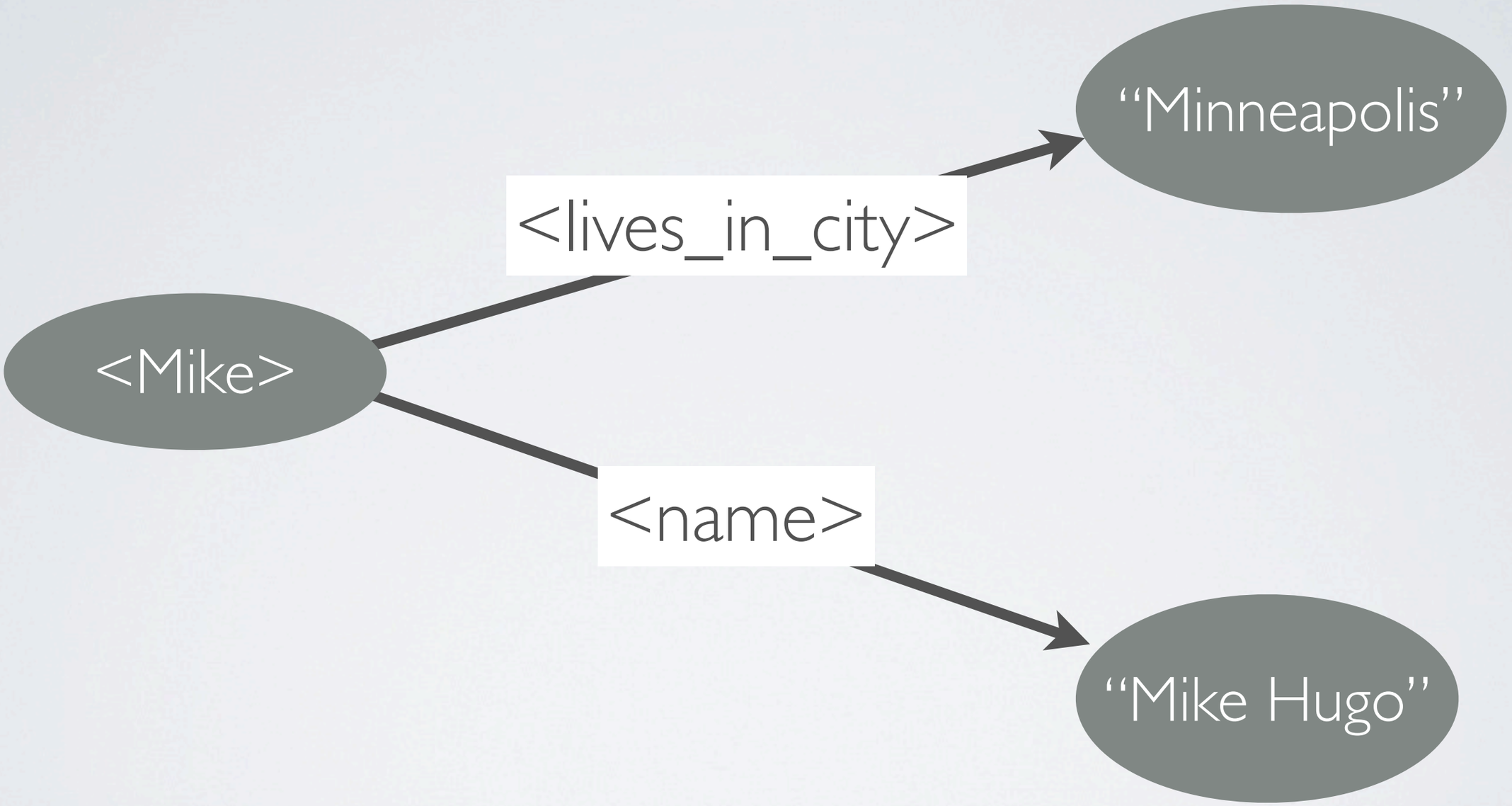



ENTAGEN
ACCELERATING INSIGHT

# WHAT'S A TRIPLE?

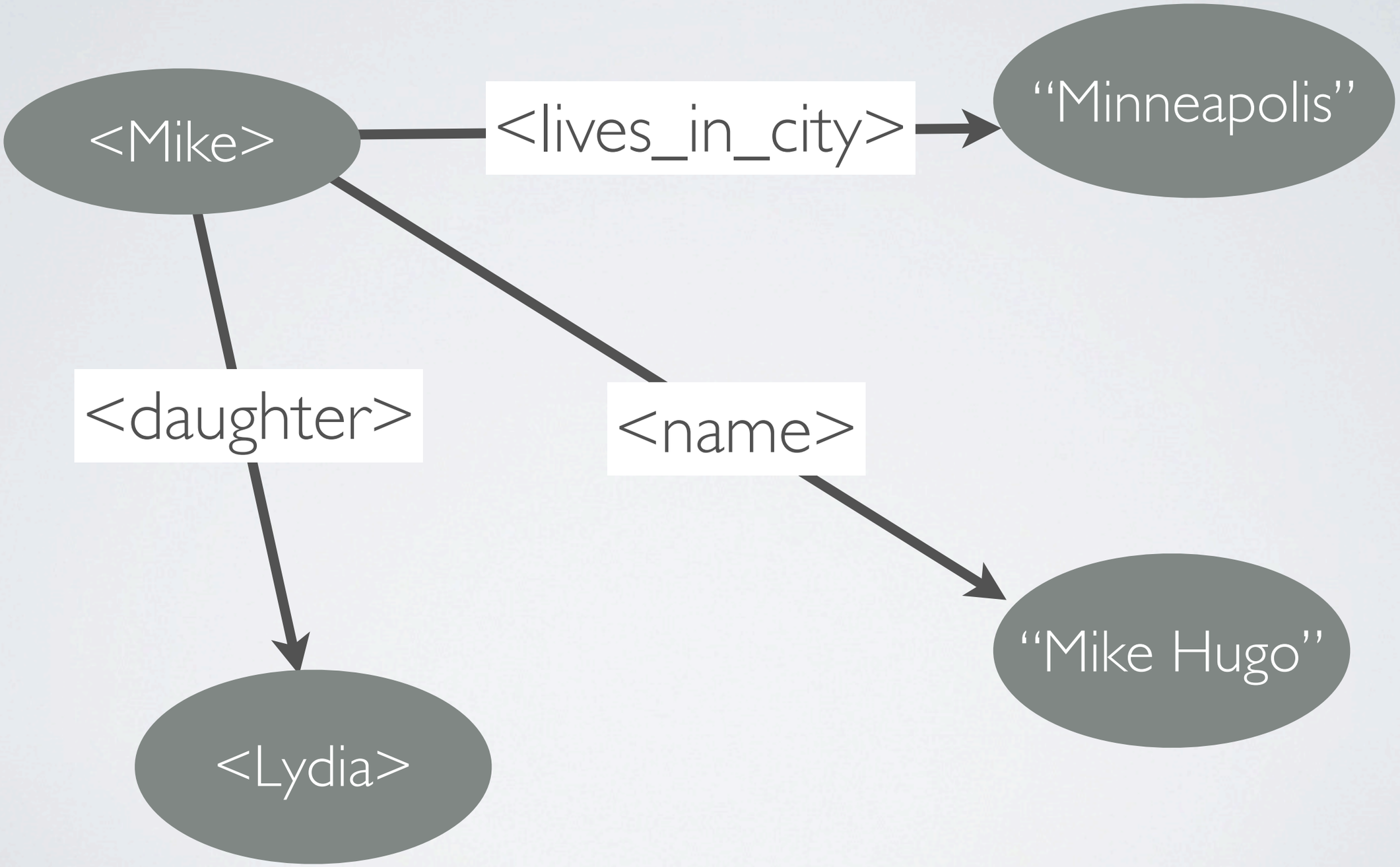


# WHAT'S A TRIPLE?



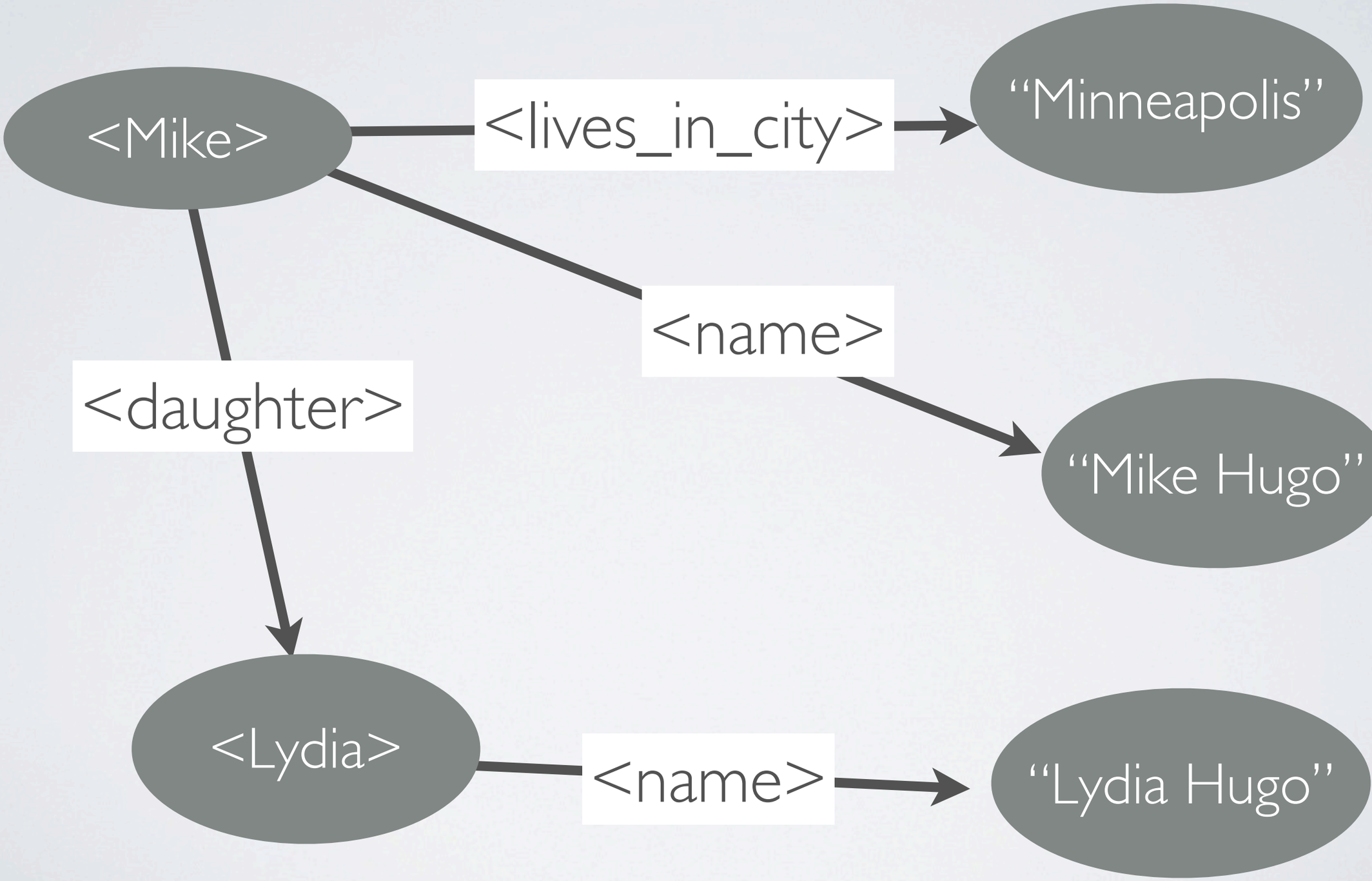

ENTAGEN
ACCELERATING INSIGHT

# WHAT'S A TRIPLE?

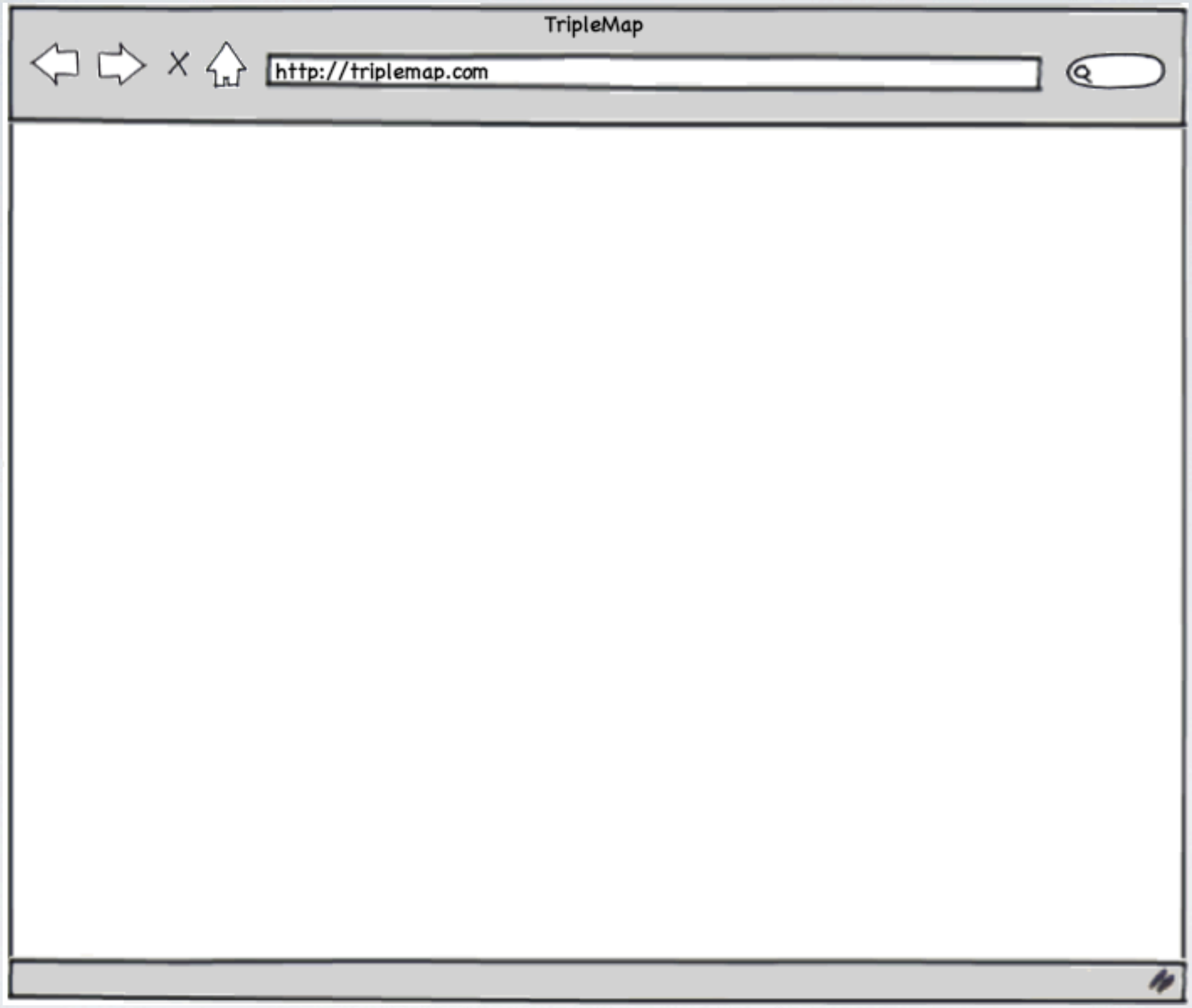



ENTAGEN
ACCELERATING INSIGHT

# WHAT'S A TRIPLE?




ENTAGEN
ACCELERATING INSIGHT




ENTAGEN
ACCELERATING INSIGHT

TripleMap

http://triplemap.com

Search

Results:

| ID | Label |
|---|---|
| 1 | Lorem |
| 2 | Ipsum |
| 3 | Dolor |
| 4 | Sit Amet |

ENTAGEN
ACCELERATING INSIGHT

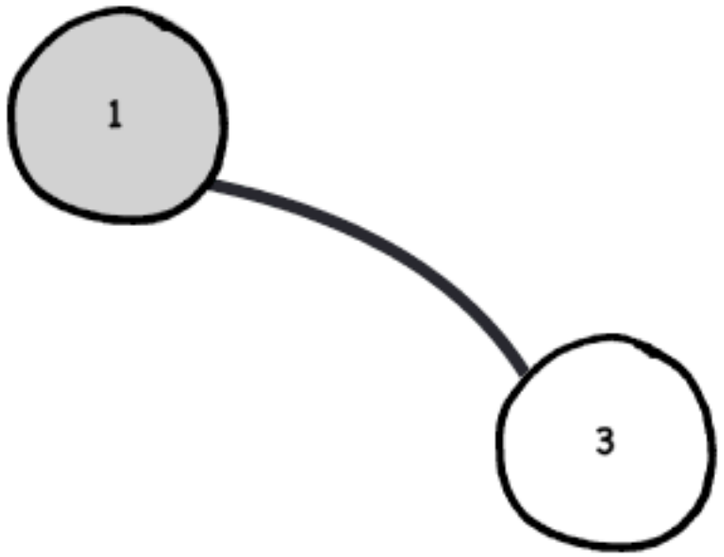
TripleMap

http://triplemap.com

Search

Related to "1"

| Related |
| --- |
| 2 |
| 4 |
| |
| |



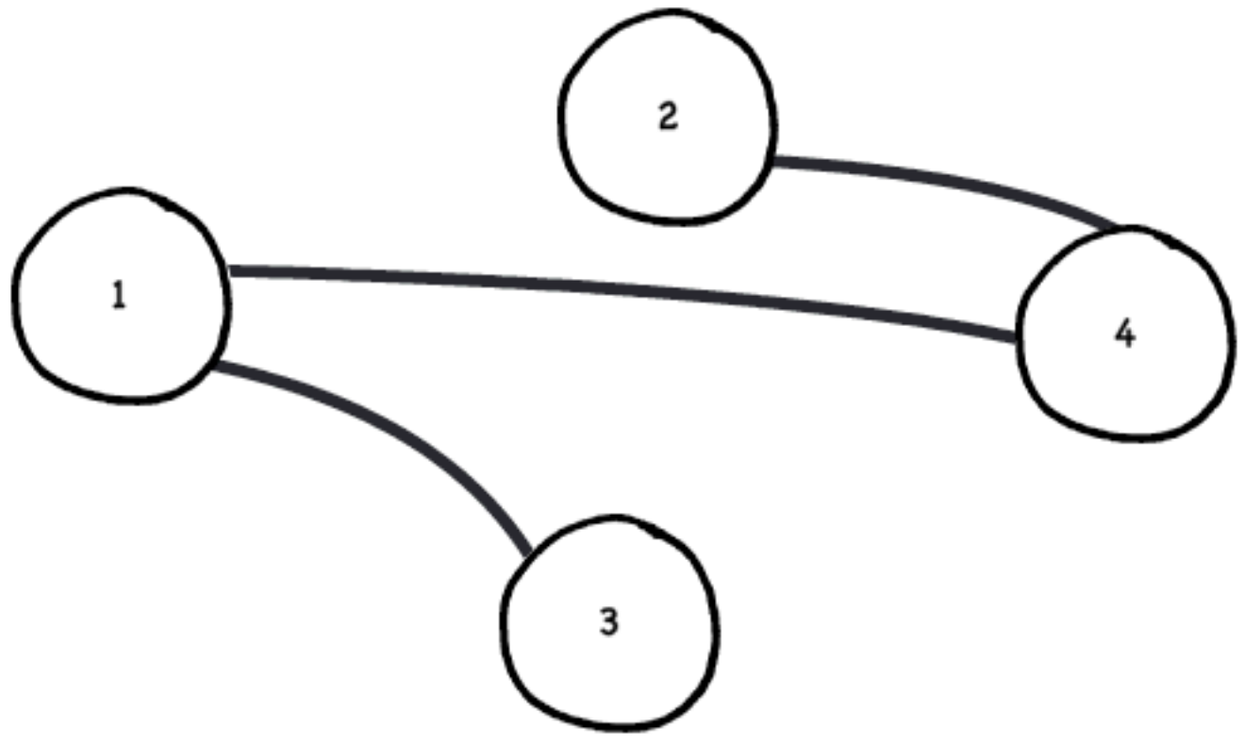

ENTAGEN
ACCELERATING INSIGHT

TripleMap

http://triplemap.com

Search

Results:




ENTAGEN
ACCELERATING INSIGHT

# Ready, GO!

ENTAGEN
ACCELERATING INSIGHT

```
select id, label
from targets
where label = '${queryValue}'
```

```
select id, label
from targets
where label
 ilike '%${queryValue}%'
```

```
SELECT ?uri ?type ?label WHERE {
   ?uri rdfs:label ?label .
   ?uri rdf:type ?type .
   FILTER (?label = '${params.query}')
} LIMIT 10
```

ENTAGEN
ACCELERATING INSIGHT

```
SELECT ?uri ?type ?label WHERE {
  ?uri rdfs:label ?label .
  ?uri rdf:type ?type .
  FILTER regex(?label,
    '\\\\Q${params.query}\\\\E', 'i')
} LIMIT 10
```

```
SELECT ?uri ?type ?label WHERE {
   ?uri rdfs:label ?label .
   ?uri rdf:type ?type .
   FILTER regex(?label,
      '\\\\Q${params.query}\\\\E', 'i')
} LIMIT 10
```

query as literal value

case insensitive

ENTAGEN
ACCELERATING INSIGHT

# DEMO

Baseline SPARQL Query Performance

ENTAGEN
ACCELERATING INSIGHT

# FASTER!

ENTAGEN
ACCELERATING INSIGHT

# Lucene


# Java API

# Indexing and Searching Text

ENTAGEN
ACCELERATING INSIGHT

lucid IMAGINATION

IBM®

AOL®

LinkedIn™

comcast®

ATLASSIAN

http://wiki.apache.org/lucene-java/PoweredBy

ENTAGEN
ACCELERATING INSIGHT

# indexing


# storage

ENTABEN

Lucene

# Document


# Document

| field | value |
|---|---|
| ID | 2 |
| name | “Mike Hugo” |
| company | “Entagen” |
| bio | “lorem ipsum dolor sum etc...” |

ENTAGEN
ACCELERATING INSIGHT

Lucene

# Index

| field | value |
|---|---|
| id | 2 |
| name | “Mike Hugo” |
| company | “Entagen” |
| bio | “lorem ipsum dolor sum etc.. |

**Indexed not Stored**

ENTAGEN
ACCELERATING INSIGHT


Query: **name: mike**

Lucene

Query: **name: mike**

Matching Documents:

| field | value |
|---|---|
| id | 2 |

ENTAGEN
ACCELERATING INSIGHT

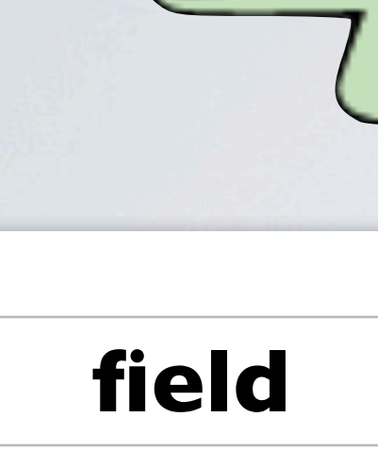
| field | value |
|---|---|
| id | 2 |

Lucene

| field | value |
| --- | --- |
| id | 2 |

Lucene

| field | value |
|---|---|
| id | 2 |



| field | value |
|---|---|
| ID | 2 |
| name | “Mike Hugo” |
| company | “Entagen” |
| bio | “lorem ipsum dolor sum etc...” |

ENTAGEN
ACCELERATING INSIGHT

# Simplest Solution

ENTAGEN
ACCELERATING INSIGHT

# Lucene index of rdfs:label

ENTAGEN
ACCELERATING INSIGHT

# Build the Index

ENTAGEN
ACCELERATING INSIGHT

```
String queryLabels = """
    SELECT ?uri ?label
    WHERE {
        ?uri rdfs:label ?label .
    }
"""
```

**Build a SPARQL query to find all the rdfs:label properties**

```
sparqlQueryService.executeForEach(repo
    def doc = new Document()
    String uri = it.uri.stringValue()
    String label = it.label.stringValu

    doc.add(new Field(SUBJECT_URI_FIE
            Field.Store.YES, Field.Ind
```

```
sparqlQueryService.executeForEach
  (repository, queryLabels) {
    String uri = it.uri.stringValue()
    String label = it.label.stringValu
```

**Execute the SPARQL query**

```
    new Document()

    doc.add(new Field(SUBJECT_URI_FIE
            Field.Store.YES, Field.In
    doc.add(new Field(LABEL_FIELD, la
            Field.Store.NO, Field.Inde

    writer.addDocument(doc)

}
```

```
iqiQueryService.executeForEach(reposito
 String uri = it.uri.stringValue()
 String label = it.label.stringValue()

 Document doc = new Document()
 doc.add(new Field(SUBJECT_URI_FIELD,
           uri,
           Field.Store.Y
           Field.Index.ANALYZED))
doc.add(new Field(LABEL_FIELD,
           label,
           Field.Store.NO,
           Field.Index.ANALYZED))

writer.addDocument(doc)
```

**Instantiate a new Lucene Document**

```
Document doc = new Document()
doc.add(new Field(SUBJECT_URI_FIELD,
        uri,
        Field.Store.YES,
        Field.Index.ANALYZED))
doc.add(new Field(LABEL_FIELD,
        label,
        Field.Store.NO
        Field.Index.AN

 writer.addDocument(doc)

ly {
```

key

value

**Add the Subject URI to the Document**

```
            Field.Store.YES,
            Field.Index.ANALYZED));
doc.add(new Field(LABEL_FIELD,
    label,
            Field.Store.NO,
            Field.Index.ANALYZED));


 writer.addDocument

ly {
ter.close()  // Close index
```

key

value

**Add the Label field to the document (but don't store it)**

ENTAGEN
ACCELERATING INSIGHT

```
                Field.Store.NO,
                Field.Index.ANALYZED))

    writer.addDocument(doc)
}
inally {
 writer.close()   //
```

**Add the document to the Index**

# Query the Index

ENTAGEN
ACCELERATING INSIGHT

```
query = {
Query query = new QueryParser(
    Version.LUCENE_CURRENT,
    LABEL_FIELD,
    new StandardAnalyzer())
            .parse(params.query);
```

query this field

for this value

```
def s                                    ime
List                                     teQuery(query)
def e                                    ime

render(view: 'index', model: [results:
```

**Create a Lucene Query from user input**

```
IndexSearcher searcher = luceneSearche
ScoreDoc[] scoreDocs =
    searcher.search(query, 10).scoreDo
List results = []
def connection = r                    ction
scoreDocs.each {
    Document doc =                    t.doc
    String uri = doc[SUBJECT_URI_FIELD
    Map labelAndType = sparqlQueryServ
    results << [uri: uri, type: labelA
}
connection.close()
return results
```

**Search the index (limit 10) for matching documents**

```
List results = []
def connection = repository.connection
scoreDocs.each {
  Document doc = searcher.doc(it.doc)
  String uri = doc[SUBJECT_URI_FIELD]
  Map labelAndType =
    spa
      g                                    i, connection)
  resu
                uri: uri,
                type: labelAndType.type,
                label: labelAndType.label])
}
connection.close()
return results
```

**For each matching document, get the doc and extract the Subject URI**

```
List results = []
def connection = repository.connection
scoreDocs.each {
  Document doc = searcher.doc(it.doc)
  String uri = doc[SUBJECT_URI_FIELD]
  Map labelAndType =
   sparqlQueryService.
     getLabelAndType(uri, connection)
  results.add([
        uri: uri
        type: la
        label: l                    ])
}
connection.close()
return results
```

**Using the Subject URI, load properties from the triplestore**

```
List results = []
def connection = repository.connection
scoreDocs.each {
  Document doc
  String uri =
  Map labelAnd
   sparqlQueryService.
     getLabelAndType(uri, connection)
  results.add([
        uri: uri,
        type: labelAndType.type,
        label: labelAndType.label])
}
connection.close()
return results
```

.doc)

IELD]

**return results containing Subject URI, Type, and Label**

# DEMO

Lucene Index of Searchable Labels

ENTAGEN
ACCELERATING INSIGHT

# WHAT ABOUT ENTITY RELATIONSHIPS?

ENTAGEN
ACCELERATING INSIGHT

# WHAT ABOUT OTHER PROPERTIES?

ENTAGEN
ACCELERATING INSIGHT

**SIREn**

*{ Semantic Information Retrieval Engine }*

# Lucene Extension

# Indexing and Searching Semi-Structured Data

ENTAGEN
ACCELERATING INSIGHT

**SIREn**

*{ Semantic Information Retrieval Engine }*

# Document

**SIREn**

*{ Semantic Information Retrieval Engine }*

# Document

| field | value |
|---|---|
| URI | <DB00619> |
| triples | <DB00619> rdfs:label "Imatinib" .<br><DB00619> rdf:type <drugbank:drugs> .<br><DB00619> drugbank:brandName "Gleevec" .<br><DB00619> drugbank:target <targets/1588> . |

ENTAGEN
ACCELERATING INSIGHT

# Build the Index

ENTAGEN
ACCELERATING INSIGHT

```
Connection connection = repository.conn
y {
  String subjectUris = """
     SELECT distinct ?uri
     WHERE {
          ?uri ?p ?o .
     }
  """

  sparqlQueryService.executeForEach(rep
       def doc = ne

       String subje                    ring
       doc.add(new Field(SUBJECT_URI_FIE
                 subjectUri,
                 Field.Store.YES
```

**Select all Subject URIs from the triplestore**

```
"""
sparqlQueryService.executeForEach(
  repository, subjectUris) {
    def doc = new Document()

    String subjectUri = it.uri.stringV
    doc.add(new Field(SUBJECT_URI_FIEL
            subjectUri,
            Field.Store.YES,

    StringWr                    er =
    NTriplesWriter nTriplesWriter =
        new NTriplesWriter(triplesStri
    connection.exportStatements(
```

**Execute the Sparql Query**
**For each URI, create a new Document**

```
def doc = new Document()

String subjectUri = it.uri.stringValue
doc.add(new Field(SUBJECT_URI_FIELD,
          subjectUri,
          Field.Store.YES,
          Field.Index.ANALYZED))

StringWriter triplesStringWriter = new
NTriplesWriter nT
    new NTriplesW                    ngWr
connection.exportStatements(
          new URIImpl(subjectUri),
          null, null, false,
```

**Add the Subject URI to the Document**

```
        Field.Index.ANALYZED))

StringWriter triplesStringWriter = new
NTriplesWriter nTriplesWriter =
    new NTriplesWriter(triplesStringWr
connection.exportStatements(
        new URIImpl(subjectUri),
        null, null, false,
        nTriplesWriter)

doc.add(new Field(T
        triplesStri                    ())
        Field.Store
        Field.Index.ANALYZED))
```

**Get an NTriples string from the triplestore**

```
            new URIImpl(subjectUri),
            null, null, false,
            nTriplesWriter)


doc.add(new Field(TRIPLES_FIELD,
            triplesStringWriter.toString(),
            Field.Store.NO,
            Field.Index.ANALYZED))

writer.addDocumen
```

**Add the NTriples string to the document**

```
                                    );

doc.add(new Field(TRIPLES_FIELD,
                  triplesStringWriter.toString(
                  Field.Store.NO,
                  Field.Index.ANALYZED))

writer.addDocument(doc)
```

**Add the document to the index**

# Query the Index

ENTAGEN
ACCELERATING INSIGHT

```
SirenCellQuery predicate =
  new SirenCellQuery(
    new SirenTermQuery(
      new Term(TRIPLES_FIELD,
      RDFS.LABEL.stringValue())));
predicate.constraint = PREDICATE_CELL

SirenCellQuery object =
  new SirenCellQue
    new SirenTermQu
      new Term(TRIPLES_FIELD,
      params.query.toLowerCase()))
object.constraint = OBJECT_CELL
```

**query the Triples field**

```
SirenCellQuery predicate =
  new SirenCellQuery(
    new SirenTermQuery(
        new Term(TRIPLES_FIELD,
        RDFS.LABEL.stringValue())));
predicate.constraint = PREDICATE_CELL
```

```
SirenCellQuery object =
  new SirenCellQue
    new SirenTermQuery(
        new Term(TRIPLES_FIELD,
        params.query.toLowerCase()))
object.constraint = OBJECT_CELL
```

**for a predicate**

```
SirenCellQuery predicate =
  new SirenCellQuery(
    new SirenTermQuery(
        new Term(TRIPLES_FIELD,
        RDFS.LABEL.stringValue())));
predicate.constraint = PREDICATE_CELL

SirenCellQuery object
  new SirenCellQuery(
    new SirenTermQuery(
        new Term(TRIPLES_FIELD,
                                Case()))
object.constraint = OBJECT_CELL
```

**of rdfs:label ***

*** note: could be any predicate!**

```
SirenCellQuery object =
   new SirenCellQuery(
    new SirenTermQuery(
                new Term(TRIPLES_FIELD,
                         params.query.toLowerCase()))
object.constraint = OBJECT_CELL

Query query = new SirenTupleQuery()
query.add(predicat
          SirenTuple                    T)
query.add(object,
          SirenTupleClause.Occur.MUST)
```

**query the Triples field**

```
SirenCellQuery object =
   new SirenCellQuery(
    new SirenTermQuery(
            new Term(TRIPLES_FIELD,
            params.query.toLowerCase()))
object.constraint = OBJECT_CELL
```

```
Query query = new SirenTupleQuery()
query.add(predicate
        SirenTupleClause.Occur.MUST)
query.add(object,
        SirenTupleClause.Occur.MUST)
```

**for an object**



```
List results = executeQuery(query)
```

```
SirenCellQuery object =
   new SirenCellQuery(
    new SirenTermQuery(
                new Term(TRIPLES_FIELD,
                  params.query.toLowerCase()))
object.constraint = OBJECT_CELL
```

```
Query query = new SirenTupleQuery()
query.add(predicat
        SirenTuple                    T)
query.add(object,
        SirenTupleClause.Occur.MUST)
```

**matching the user input**


```
List results = executeQuery(query)
```

| field | value |
|---|---|
| URI | <DB00619> |
| triples | <DB00619> rdfs:label "Imatinib" .<br><DB00619> rdf:type <drugbank:drugs> .<br><DB00619> drugbank:brandName "Gleevec" .<br><DB00619> drugbank:target <targets/1588> . |

Query: **“imatinib”**

ENTAGEN
ACCELERATING INSIGHT

| field | value |
| --- | --- |
| URI | <DB00619> |
| triples | <DB00619> rdfs:label "Imatinib" .<br><DB00619> rdf:type <drugbank:drugs> .<br><DB00619> drugbank:brandName "Gleevec" .<br><DB00619> drugbank:target <targets/1588> . |

Query:

**triples field**

| field | value |
|---|---|
| URI | <DB00619> |
| triples | <DB00619> rdfs:label "Imatinib" .<br><DB00619> rdf:type <drugbank:drugs> .<br><DB00619> drugbank:brandName "Gleevec" .<br><DB00619> drugbank:target <targets/1588> . |

Query:

**predicate = rdfs:label**

ENTAGEN
ACCELERATING INSIGHT

| field | value |
| --- | --- |
| URI | <DB00619> |
| triples | <DB00619> rdfs:label "Imatinib".<br><DB00619> rdf:type <drugbank:drugs> .<br><DB00619> drugbank:brandName "Gleevec" .<br><DB00619> drugbank:target <targets/1588> . |

Query:

**predicate = rdfs:label**

**object = “imatinib”**

ENTAGEN
ACCELERATING INSIGHT

```
List executeQuery(Query query) {
IndexSearcher searcher = sirenSearcherM
ScoreDoc[] scoreDocs =
  searcher.search(query, 10).scoreDocs
List results = []
def connection = repos                    n
scoreDocs.each {
    Document doc = sea                    c
    String uri = doc[S                    D
    Map labelAndType = sparqlQueryServi
        getLabelAndType(uri, connectio
    results.add([
            uri: uri,
            type: labelAndType.type,
```

**Search the index (limit 10) for matching documents**

```
List results = []
def connection = repository.connection
scoreDocs.each {
    Document doc = searcher.doc(it.doc)
    String uri = doc[SUBJECT_URI_FIELD]
    Map labelAndType = sparqlQueryServic
            get                    connectio
    results.
                                   pe.type,
                label: labelAndType.label])
}

connection.close()
return results
```

**For each matching document, get the doc and extract the Subject URI**

```
 connection = repository.connection
reDocs.each {
 Document doc = searcher.doc(it.doc)
 String uri = doc[SUBJECT_URI_FIELD]
 Map labelAndType = sparqlQueryService.
       getLabelAndType(uri, connection)
 results.add([
              uri: uri,
              type: lab
              label: la
nection.close()
urn results
```

**Using the Subject URI, load properties from the triplestore**

ENTAGEN
ACCELERATING INSIGHT

```
String uri = doc[SUBJECT_URI_FIELD]
Map labelAndType = sparqlQueryService.
      getLabelAndType(uri, connection)
results.add([
        uri: uri,
        type: labelAndType.type,
        label: labelAndType.label])
```

```
nection.close()
urn results
```

**return results containing Subject URI, Type, and Label**

# DEMO

SIREn Index of RDF Entities

ENTAGEN
ACCELERATING INSIGHT

# FLEXIBILITY

ENTAGEN
ACCELERATING INSIGHT

| field | value |
| --- | --- |
| URI | <DB00619> |
| triples | <DB00619> rdfs:label "Imatinib".<br><DB00619> rdf:type <drugbank:drugs> .<br><DB00619> drugbank:brandName "Gleevec" .<br><DB00619> drugbank:target <targets/1588> . |

Query:

**predicate = rdfs:label**

**object = “imatinib”**

ENTAGEN
ACCELERATING INSIGHT

| field | value |
|---|---|
| URI | <DB00619> |
| triples | <DB00619> rdfs:label "Imatinib".<br><DB00619> rdf:type <drugbank:drugs> .<br><DB00619> drugbank:brandName "Gleevec" .<br><DB00619> drugbank:target <targets/1588> . |

Query:

**object = “imatinib”**

ENTAGEN
ACCELERATING INSIGHT

| field | value |
| --- | --- |
| URI | <DB00619> |
| triples | <DB00619> rdfs:label "Imatinib".<br><DB00619> rdf:type <drugbank:drugs> .<br><DB00619> drugbank:brandName "Gleevec"<br><DB00619> drugbank:target <targets/1588> . |

Query:

**object = “imatinib”**

**OR**

**object = “gleevec”**

ENTAGEN
ACCELERATING INSIGHT

# MORE THAN LITERALS

ENTAGEN
ACCELERATING INSIGHT

| field | value |
|---|---|
| URI | <DB00619> |
| triples | <DB00619> rdfs:label "Imatinib" .<br><DB00619> rdf:type <drugbank:drugs> .<br><DB00619> drugbank:brandName "Gleevec" .<br><DB00619> drugbank:target <targets/1588> . |

Query:

**predicate = brandName**

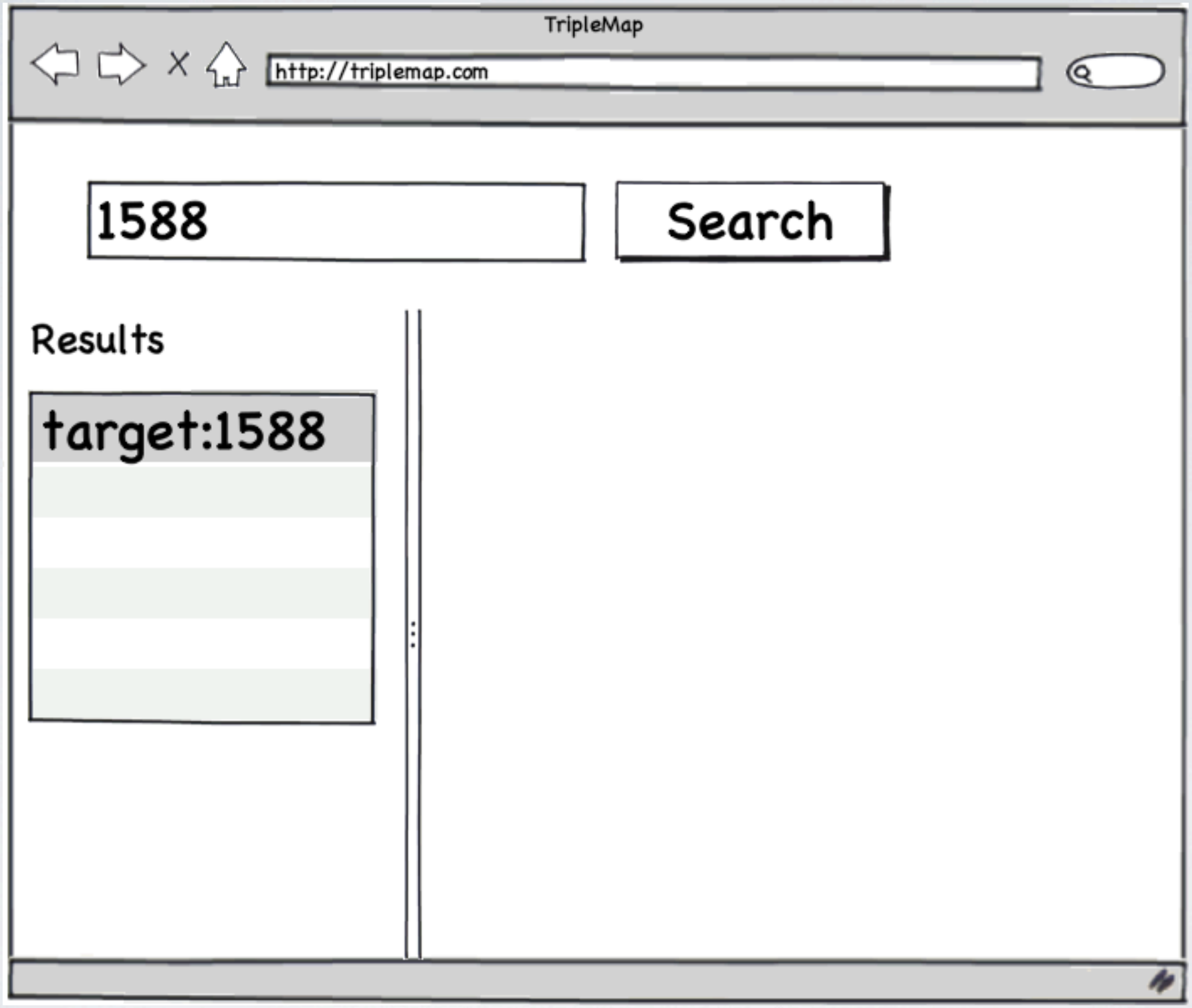
| field | value |
| --- | --- |
| URI | <DB00619> |
| triples | <DB00619> rdfs:label "Imatinib" .<br><DB00619> rdf:type <drugbank:drugs> .<br><DB00619> drugbank:brandName "Gleevec" .<br><DB00619> drugbank:target <targets/1588> . |

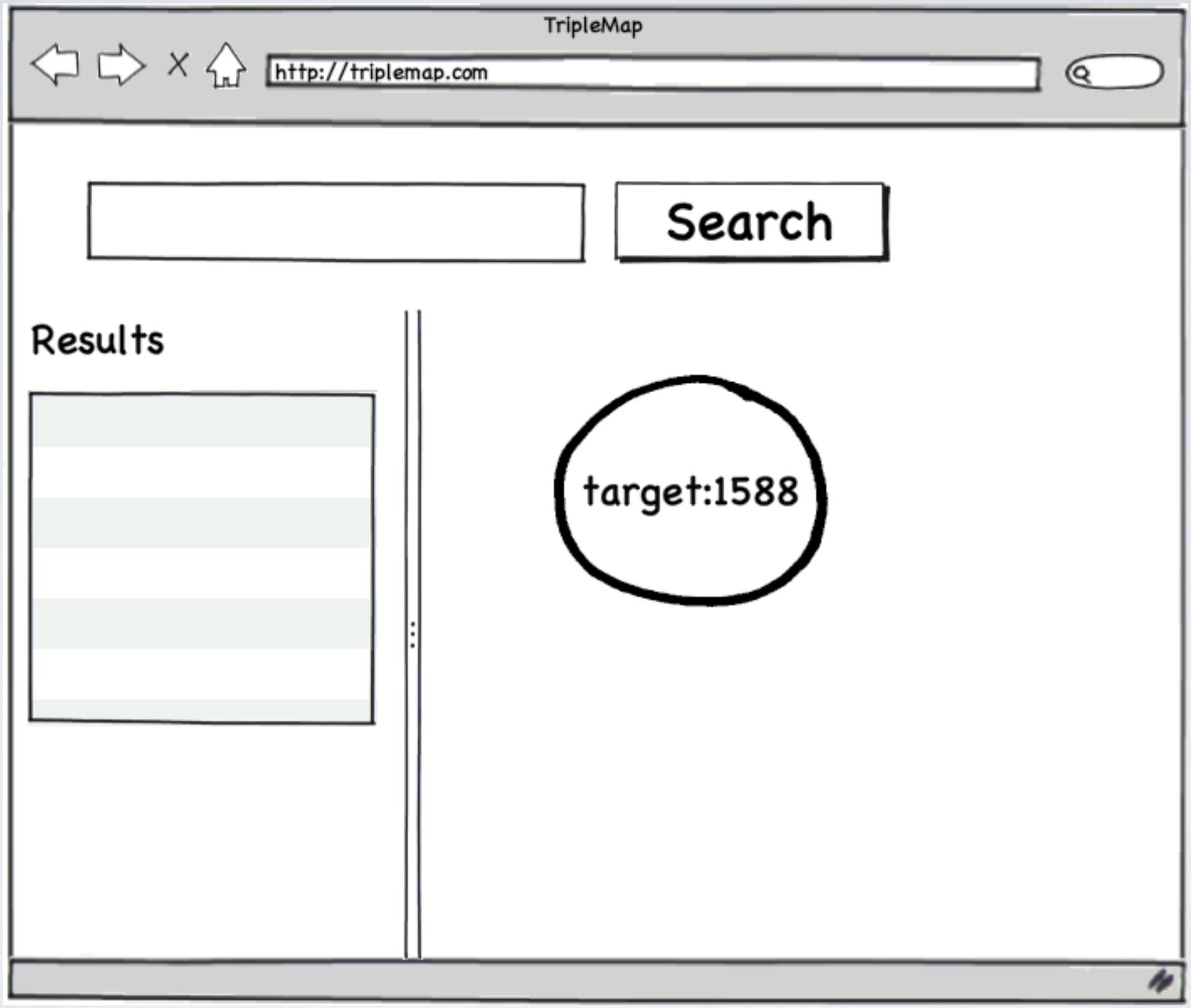
Query:

**predicate = target**

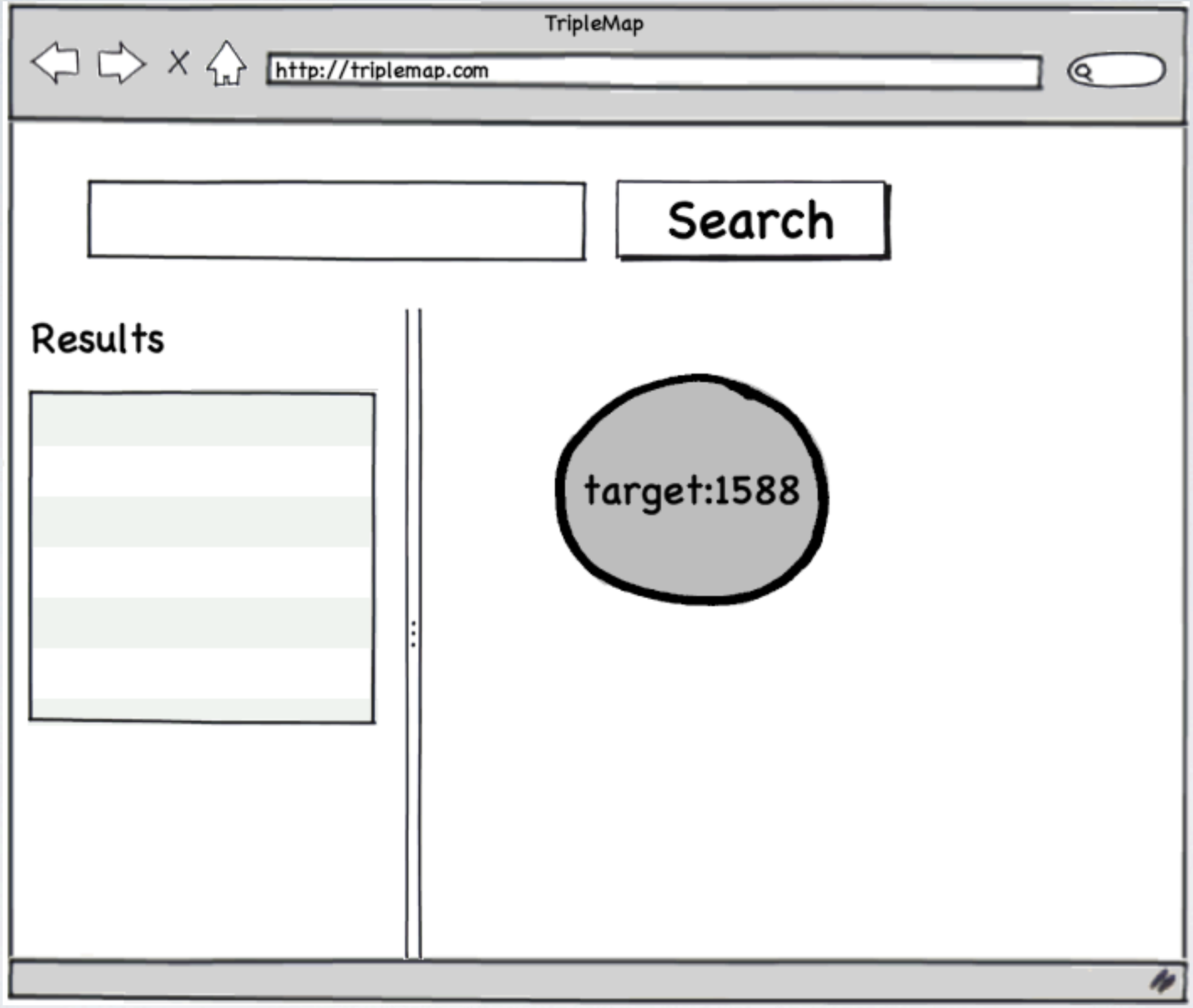
# RELATIONSHIPS

ENTAGEN
ACCELERATING INSIGHT


TripleMap
http://triplemap.com
1588
Search
Results
target:1588




ENTAGEN
ACCELERATING INSIGHT




ENTAGEN
ACCELERATING INSIGHT

TripleMap

http://triplemap.com

Search

Related to 1588:

| DB00619 |
| --- |
| |
| |
| |
| |
| |


ENTAGEN
ACCELERATING INSIGHT

| field | value |
| --- | --- |
| URI | <DB00619> |
| triples | <DB00619> rdfs:label "Imatinib" .<br><DB00619> rdf:type <drugbank:drugs> .<br><DB00619> drugbank:brandName "Gleevec" .<br><DB00619> drugbank:target <targets/1588> . |

Query:

**object = <targets/1588>**

ENTAGEN
ACCELERATING INSIGHT

# DEMO

Searching SIREn Index for Relationships

ENTAGEN
ACCELERATING INSIGHT

# SIREn

*{ Semantic Information Retrieval Engine }*

Apache Solr

# Distributed
# Indexing and Searching Semi-Structured Data

ENTAGEN
ACCELERATING INSIGHT

Apache Solr

ENTAGEN
ACCELERATING INSIGHT


Apache Solr

Apache Solr

ENTAGEN
ACCELERATING INSIGHT

ENTAGEN
ACCELERATING INSIGHT

ENTAGEN
ACCELERATING INSIGHT

indice
THE SEMANTIC WEB INDEX

Home About Se

Search interface type: Simple Advanced **Guru** Query Language Documentation

keyword(s)

ntriple(s) * <name> "Tim Berners Lee"

filter(s)

SEARCH Group By Dataset: Sorted by: relevance

Quick filters (All options)

Time range:

***Any date***

Today Yesterday Last week

Last month Last year

Format:

***Any format***

RDF RDFA MICROFORMAT XFN

HCARD HCALENDAR HLISTING

HRESUME LICENSE GEO ADR

**Sindice search:* <name> "Tim Berners Lee"** found 699 documer

http://dig.csail.mit.edu/2007/01/camp/data (RDF)

+ 2009-08-12 - 387 triples in 43.2 kB

http://dig.csail.mit.edu/2007/01/camp/data (Search) Inspect: (Cache) (Live)

http://dbpedia.org/data/Tim_Berners-Lee (RDF)

− 2011-05-18 - 171 triples in 38.8 kB

SIGMA FULL CONTENT GRAPH TRIPLES(890)

See all we know about "Tim Ber

indice
THE SEMANTIC WEB INDEX

Home About Se

Search interface type: Simple Advanced **Guru** Query Language Documentation

keyword(s)

ntriple(s) * <name> "Tim Berners Lee"

# 400 Million Documents
# > 12 Billion Triples

SEARCH Group By Dataset: Sorted by: relevance

Quick filters (All options) Sindice search: <name> "Tim Berners Lee" found 699 documen

**Time range:**

***Any date***

Today Yesterday Last week

Last month Last year

**Format:**

***Any format***

RDF RDFA MICROFORMAT XFN

HCARD HCALENDAR HLISTING

HRESUME LICENSE GEO ADR

http://dig.csail.mit.edu/2007/01/camp/data (RDF)

2009-08-12 - 387 triples in 43.2 kB

http://dig.csail.mit.edu/2007/01/camp/data (Search) Inspect: (Cache) (Live)

http://dbpedia.org/data/Tim_Berners-Lee (RDF)

2011-05-18 - 171 triples in 38.8 kB

SIGMA FULL CONTENT GRAPH TRIPLES(890)

See all we know about "Tim Ber

# Query Parser

indice
THE SEMANTIC WEB INDEX

Home About Se

Search interface type: Simple Advanced **Guru** Query Language Documentation

keyword(s)

ntriple(s) * <name> "Tim Berners Lee"

filter(s)

SEARCH Group By Dataset: Sorted by: relevance

Quick filters (All options)

Time range:

***Any date***

Today Yesterday Last week

Last month Last year

Format:

**Sindice search:* <name> "Tim Berners Lee"** found 699 documer

http://dig.csail.mit.edu/2007/01/camp/data (RDF)

+ 2009-08-12 - 387 triples in 43.2 kB

http://dig.csail.mit.edu/2007/01/camp/data (Search) Inspect: (Cache) (Live)

http://dbpedia.org/data/Tim_Berners-Lee (RDF)

− 2011-05-18 - 171 triples in 38.8 kB

nterface type: Simple Advanced **Guru** Query Language Documentation

keyword(s)

ntriple(s) * <name> "Tim Berners Lee"

filter(s)

SEARCH Group By Dataset: Sorted by: relevance

**subject** **predicate** **object**

tions)

Sindice search:* <name> "Tim Berners Lee" foun

http://dig.csail.mit.edu/2007/01/camp/data

2009-08-12 - 387 triples in 43.2 kB

http://dig.csail.mit.edu/2007/01/camp/data (Search) Inspe

st week

ar

# DEMO

SIREn in action on TripleMap.com

ENTAGEN
ACCELERATING INSIGHT

TripleMap

Welcome, Michael Hugo | **Logout** | **Help & Feedback**

**Search** Maps Analyze

Search... Search

Filter

Sort

Select Total count: 2

< Back **Indication**

Colchicine resistance

Colchicine_resistance

18

144

2

16

**Map View**

Remove

Remove All

Group

ABCB1

Proto-oncogene tyrosine-protein kinase ABL1

AssociatedGene

Target

Target

Colchicine resistance

PossibleDrug

PossibleDiseaseTarget

Imatinib

ENTAGEN
ACCELERATING INSIGHT

# SPARQL

# LUCENE

# SIREN

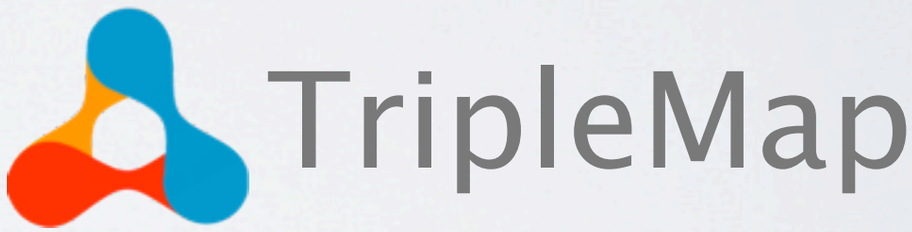
# TripleMap.com

ENTAGEN
ACCELERATING INSIGHT

# QUESTIONS?

mike@entagen.com / twitter: @piragua


http://www.entagen.com




http://www.triplemap.com