A Performance and Scalability Metric for Virtual RDF Graphs

Michael Hausenblas, Wolfgang Slany & Danny Ayers

A Retrospect

Have you EVER used a relational database (as Oracle, MySQL, etc.)?

Unsurprisingly, the 90%-answer is:

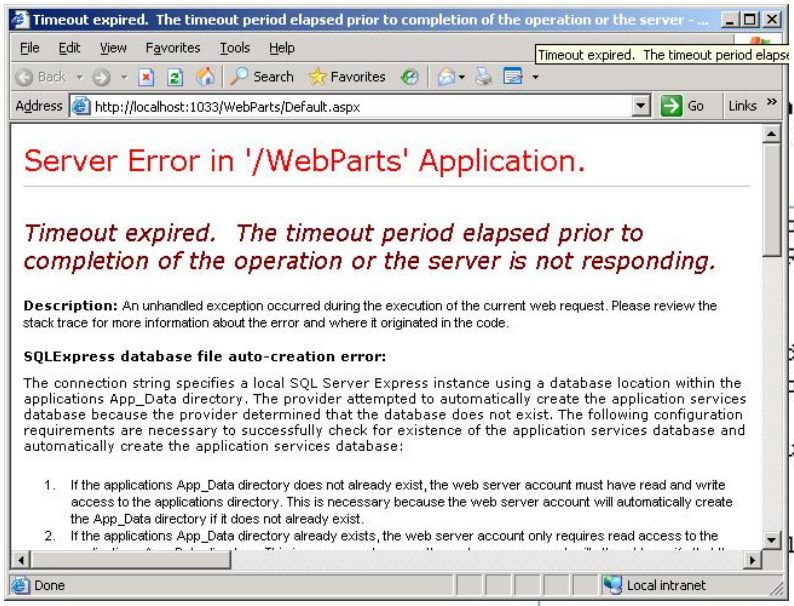
YES!

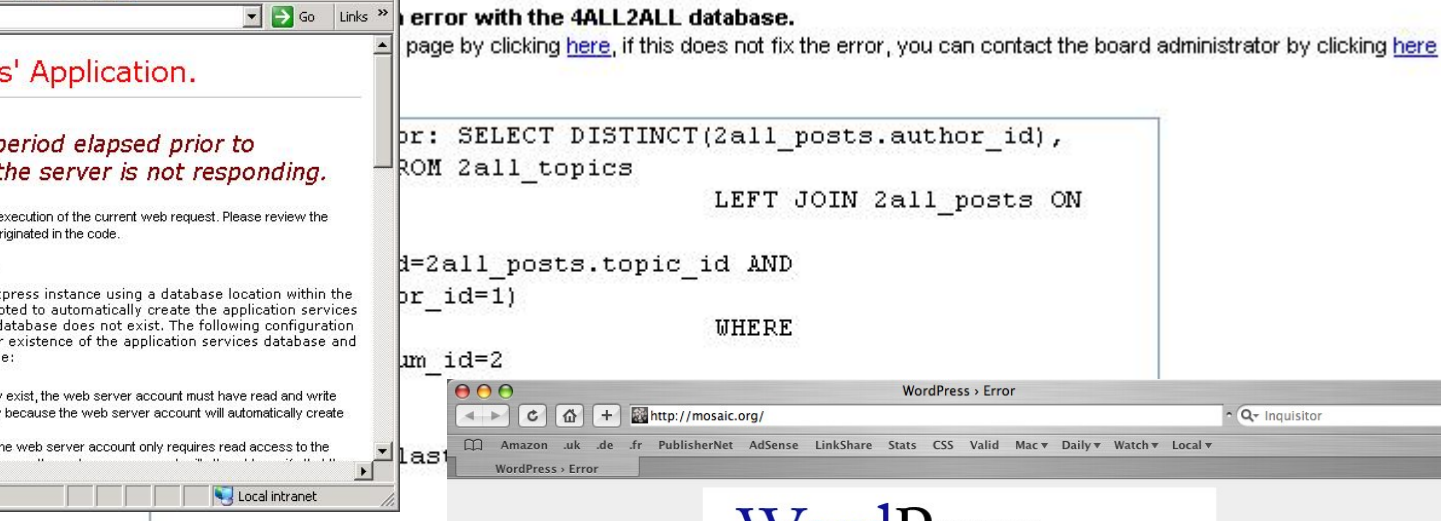
A Retrospect

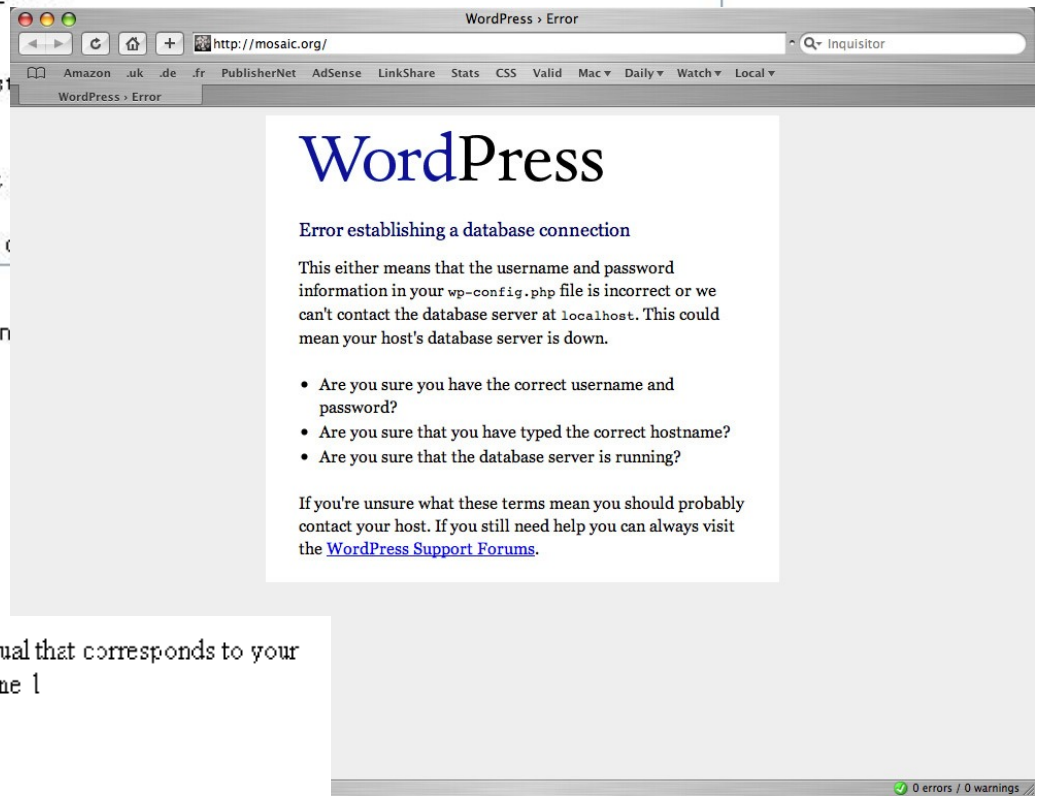
- In case YES – What do you associate with an RDBMS?
  - **mature**, industry standard (Web applications, desktop applications, etc.)
  - **standardised query and manipulation language** (SQL)
  - **reliability** (recovery, backup)
  - **transactions** (ACID)
  - ***scalability*** *&* ***performance***

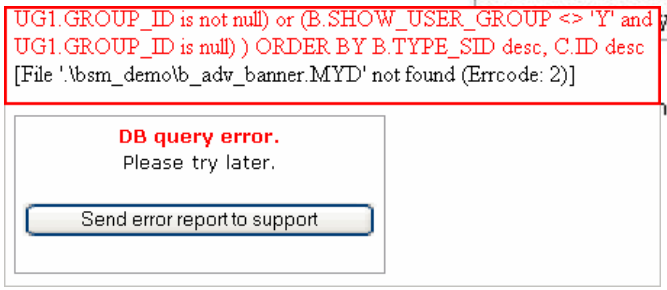
A Retrospect

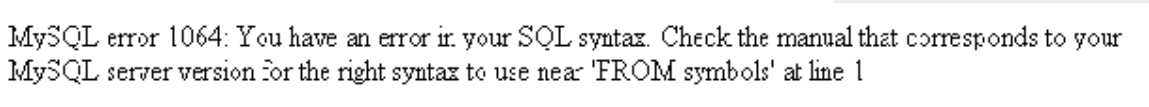
- Even if the answer would be NO, I never came across an RDBMS ...

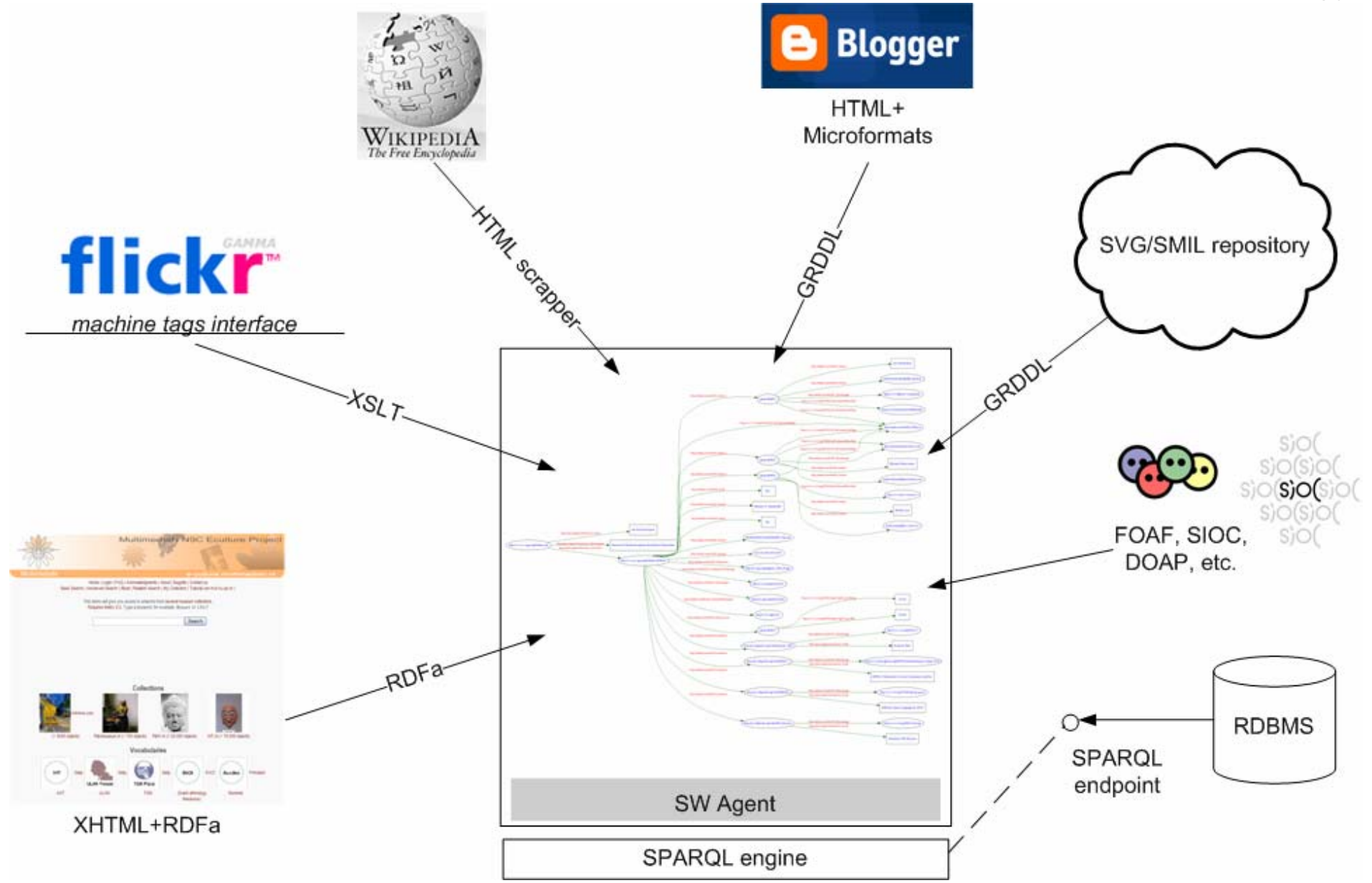

Timeout expired. The timeout period elapsed prior to completion of the operation or the server - ...
File Edit View Favorites Tools Help
Address http://localhost:1033/WebParts/Default.aspx
Server Error in '/WebParts' Application.
Timeout expired. The timeout period elapsed prior to completion of the operation or the server is not responding.
Description: An unhandled exception occurred during the execution of the current web request. Please review the stack trace for more information about the error and where it originated in the code.
SQLExpress database file auto-creation error:
The connection string specifies a local SQL Server Express instance using a database location within the applications App_Data directory. The provider attempted to automatically create the application services database because the provider determined that the database does not exist. The following configuration requirements are necessary to successfully check for existence of the application services database and automatically create the application services database:
1. If the applications App_Data directory does not already exist, the web server account must have read and write access to the applications directory. This is necessary because the web server account will automatically create the App_Data directory if it does not already exist.
2. If the applications App_Data directory already exists, the web server account only requires read access to the
Done
Local intranet


error with the 4ALL2ALL database.
page by clicking here, if this does not fix the error, you can contact the board administrator by clicking here
or: SELECT DISTINCT(2all_posts.author_id),
ROM 2all_topics
LEFT JOIN 2all_posts ON
d=2all_posts.topic_id AND
or_id=1)
WHERE
um_id=2
last


WordPress › Error
http://mosaic.org/
WordPress
Error establishing a database connection
This either means that the username and password information in your wp-config.php file is incorrect or we can't contact the database server at localhost. This could mean your host's database server is down.
Are you sure you have the correct username and password?
Are you sure that you have typed the correct hostname?
Are you sure that the database server is running?
If you're unsure what these terms mean you should probably contact your host. If you still need help you can always visit the WordPress Support Forums.


mySQL error: Can't
mySQL error code:
UG1.GROUP_ID is not null) or (B.SHOW_USER_GROUP <> 'Y' and
UG1.GROUP_ID is null)) ORDER BY B.TYPE_SID desc, C.ID desc
[File '.\bsm_demo\b_adv_banner.MYD' not found (Errcode: 2)]
DB query error.
Please try later.
Send error report to support


MySQL error 1064: You have an error in your SQL syntax. Check the manual that corresponds to your MySQL server version for the right syntax to use near 'FROM symbols' at line 1


WIKIPEDIA
The Free Encyclopedia
Blogger
HTML+
Microformats
flickr
machine tags interface
HTML scrapper
GRDDL
SVG/SMIL repository
XSLT
GRDDL
FOAF, SIOC,
DOAP, etc.
RDFa
XHTML+RDFa
SW Agent
SPARQL engine
SPARQL
endpoint
RDBMS

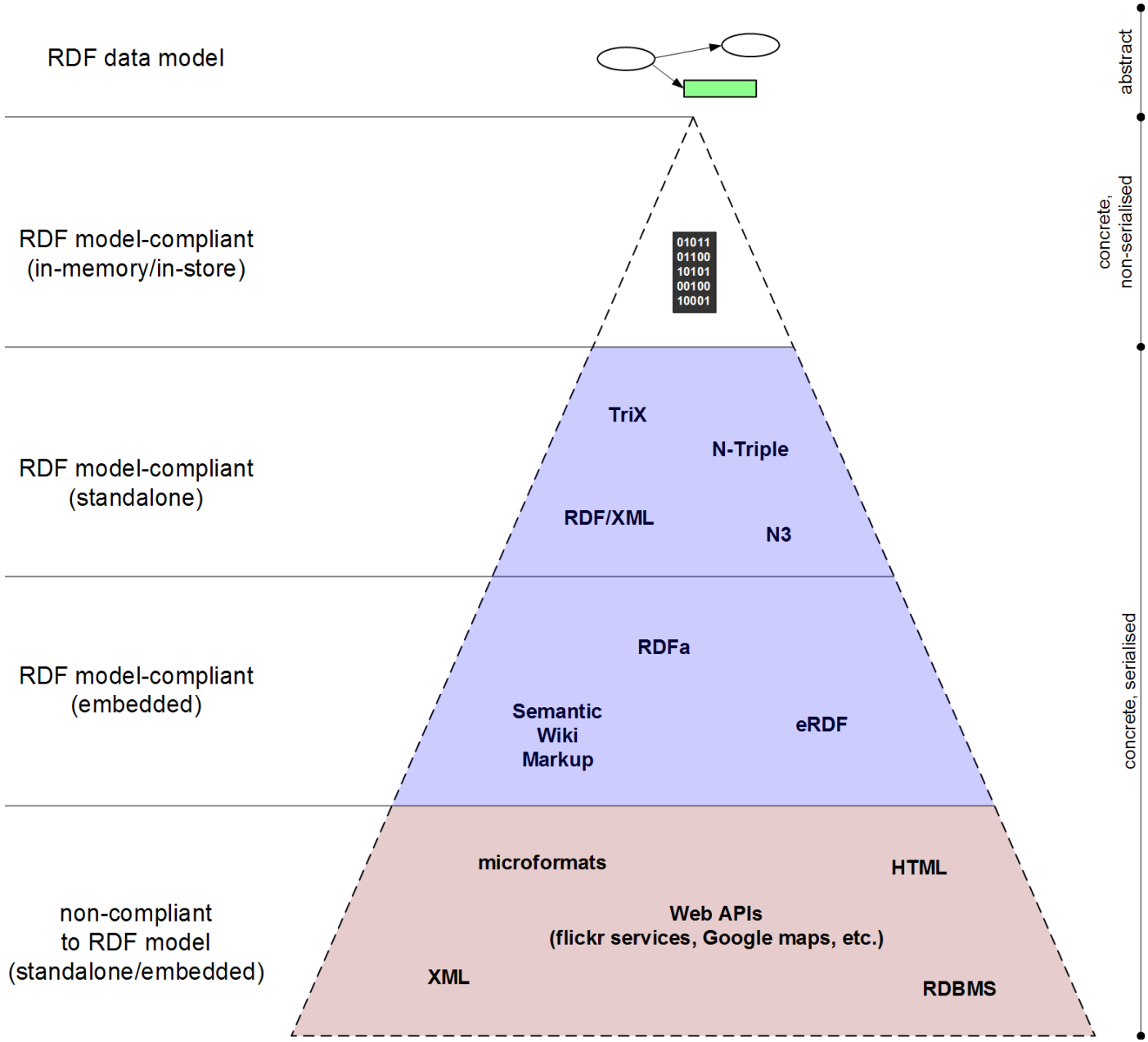
Semantic Web Application

A software program that meets the following minimal requirements:

1. It is based **HTTP** and **URI**s;
2. For human agents, the primary presentation format is (X)**HTML**, for software agents, the primary interface formats are based on Web services and/or based on the REST approach;
3. The application operates on the Internet; the number of concurrent users is undetermined.
4. The content used is machine readable and interpretable; the data model of the application is **RDF**.
5. A set of formal vocabularies (potentially based on **OWL**) is used to capture the domain of discourse. At least one of the utilised vocabularies has to be proven not to be under full control of the Semantic Web application developer.
6. *Non-mandatory:* SPARQL is in use for querying, and RIF for representing, respectively exchanging rules.

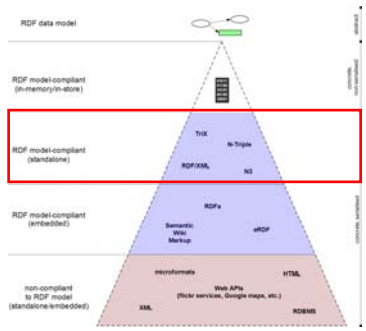
Virtual RDF graphs

- Triples may stem from sources that utilise various representations.
- Let us visualise the representational properties of the sources ranging from the RDF model to non model-compliant sources.


RDF data model
RDF model-compliant
(in-memory/in-store)
01011
01100
10101
00100
10001
RDF model-compliant
(standalone)
TriX
N-Triple
RDF/XML
N3
RDF model-compliant
(embedded)
RDFa
Semantic
Wiki
Markup
eRDF
non-compliant
to RDF model
(standalone/embedded)
microformats
HTML
Web APIs
(flickr services, Google maps, etc.)
XML
RDBMS
abstract
concrete,
non-serialised
concrete, serialised

Types Of Sources

- **Standalone, RDF model-compliant Representations**. These type of sources, for example RDF/XML, can be stored, transmitted, and processed on their own. For example an in-memory Document Object Model (DOM) representation of an RDF/XML document can be built by utilising a SAX parser.

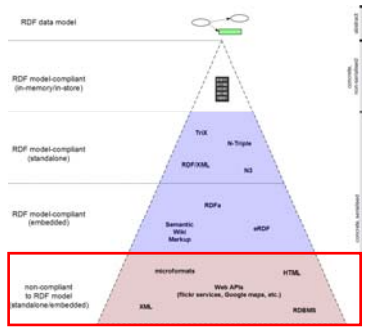


Types Of Sources

- **Embedded, RDF model-compliant Representations**. Sources of this type, as RDFa or eRDF, need a host to exist; their representation is only defined in the context of this host. Here, the triples are produced by applying a transformation.

Types Of Sources

- **Representations non-compliant to the RDF model**. The majority of the data sources on the Web, standalone or embedded, is of this type:
  - GRDDL is utilised to 'uncover' RDF in, e.g., HTML. The same applies to microformats that can be RDF-ized using hGRDDL;
  - An RDBMS that provides for a SPARQL end point can be used to contribute triples;
  - Syndicated feeds (RSS 2.0, Atom)
  - From a HTML page without explicit metadata, triples may be gathered through screen scrapers.



Characteristics Of Sources

- Fixed sized sources
  - A Wiki site may serve as a source for a vRDF graph; an HTML scraper is used to generate triples from selected Wiki pages, for example based on a category. The number of resulting triples is in many cases stable and can be assessed in advanced.
- Dynamic—sized sources
  - An RDBMS that provides for a SPARQL end point (eg D2R Server). Based on the query, the number of triples varies.
- Border cases
  - Social media sites, as blogs. They are less dynamic than data provided by a SPARQL end point but constantly changing and growing as more comments come in.

A Metric for virtual RDF Graphs

- tP overall execution time for performing a CRUD function
- tO time for the actual operation performed locally
- t2RDF the time for converting a non-RDF source representation into an RDF graph

A Metric for virtual RDF Graphs

$$tP = tO + t2RDF + tD$$

$$coR = t2RDF/tP$$

Showcase: PSIMeter

Demonstrates the application of the metric by RDF-izing the flickr API. Three different methods have been implemented; the non-native RDF Source used in the **PSIMeter** showcase is the information present in the machine tags. The goal for each of the three methods is to allow a Semantic Web agent to perform a SPARQL construct statement as:

```
CONSTRUCT {
  ?photoURL dc:subject ?subject }
WHERE {
  ?photoURL dc:subject ?subject.
  FILTER regex(?subject, "XXX", "i")
}
```

PSIMeter-Showcase: RDF-ize the flickr API

This is a demo to show alternative ways to RDF-ize the flickr API regarding machine tags. Three approaches are used to achieve the same result: to create an RDF graph that contains the requested property along with the specified value.

Query

Execute Reset

Flickr User ID

75381465@N00

List flickr users that use machine tags ...

Search for photos tagged with

dc:date =

Select execution method

☑ **Approach A** (may take some 10s) ... uses the flickr API to retrieve photos of a user tagged with a certain machine tag, and adds the used tags to each photo, finally converting the result into RDF using a local XSLT.

☐ **Approach B** (may take some 50s) ... uses the flickr API to retrieve all public photos of a user, and uses local XSL transformations to generate the RDF graph;

☐ **Approach C** (may take some 2min) ... uses the flickr API to retrieve all public photos of a user. Then, for each photo an external service is invoked to generate the RDF graph.

Output result in N3

Results

Approach A

Metrics

Overall Execution Time (tP) [ms]	Conversion Time (t2RDF) [ms]	Operation Time (tO) [ms]
5718.0	5708.0	10.0

Operation vs. Overall Execution Time Ratio (ooR)	Conversion vs. Overall Execution Time Ratio (coR)	Number of Triples (T)
0.0017488633	0.99825114	129

Operation Result

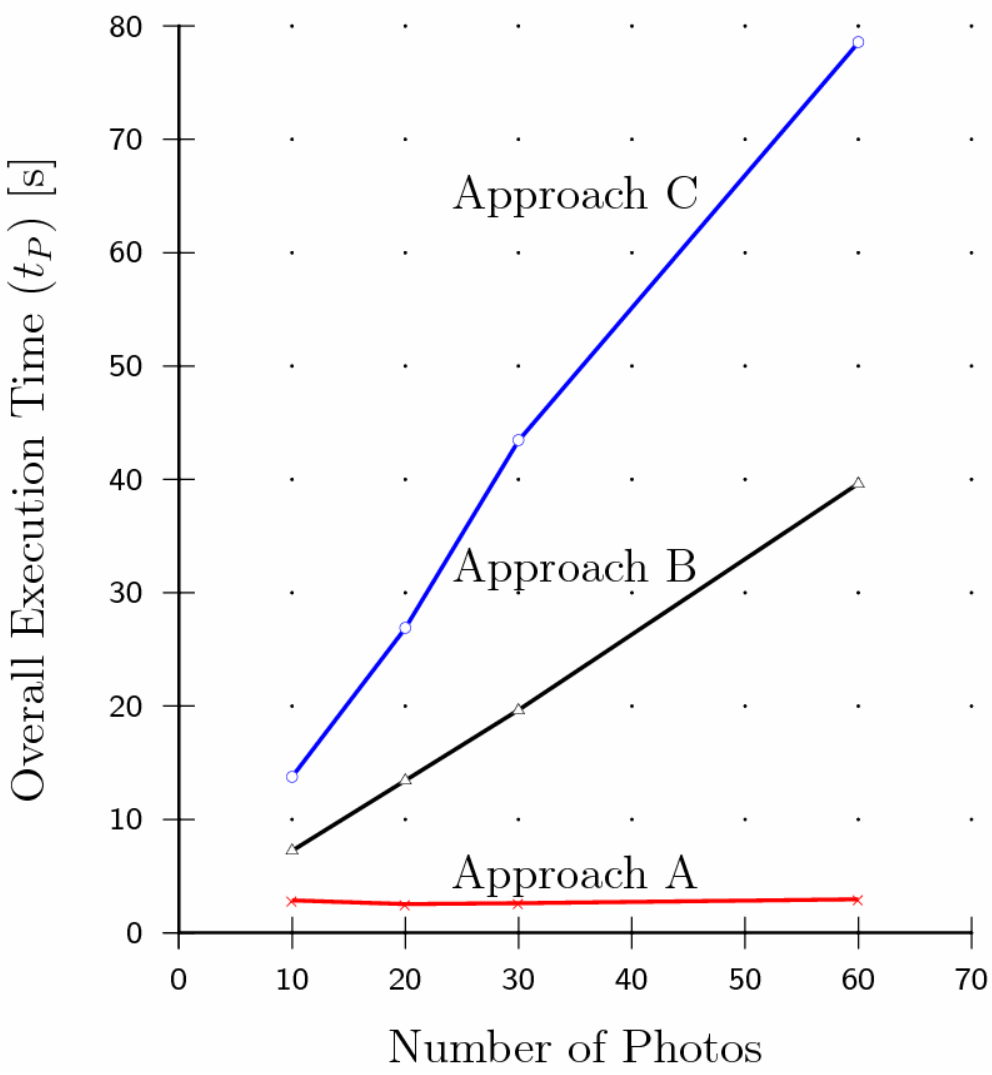
```
@prefix sxsw:     <http://sxsw.com/> .
@prefix upcoming:  <http://upcoming.org/> .
@prefix filtr:    <http://example.org/filtr#> .
@prefix xsd:      <http://www.w3.org/2001/XMLSchema#> .
@prefix cell:     <http://machinetags.org/wiki/Cell/> .
@prefix address:  <http://example.org/address#> .
@prefix ph:       <http://example.org/ph#> .
@prefix flickr:   <http://flickr.com/tags/meta#> .
@prefix rdfs:     <http://www.w3.org/2000/01/rdf-schema#> .
@prefix foaf:     <http://xmlns.com/foaf/0.1/> .
@prefix dcterms:  <http://purl.org/dc/terms/> .
@prefix geo:      <http://www.w3.org/2003/01/geo/wgs84_pos#> .
@prefix dc:       <http://purl.org/dc/elements/1.1/> .
@prefix location:  <http://example.org/location#> .
@prefix rdf:      <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix people:   <http://example.org/people#> .
@prefix :         <#> .

<http://www.flickr.com/photos/75381465@N00/503392136>
      dc:date "1988" .
```

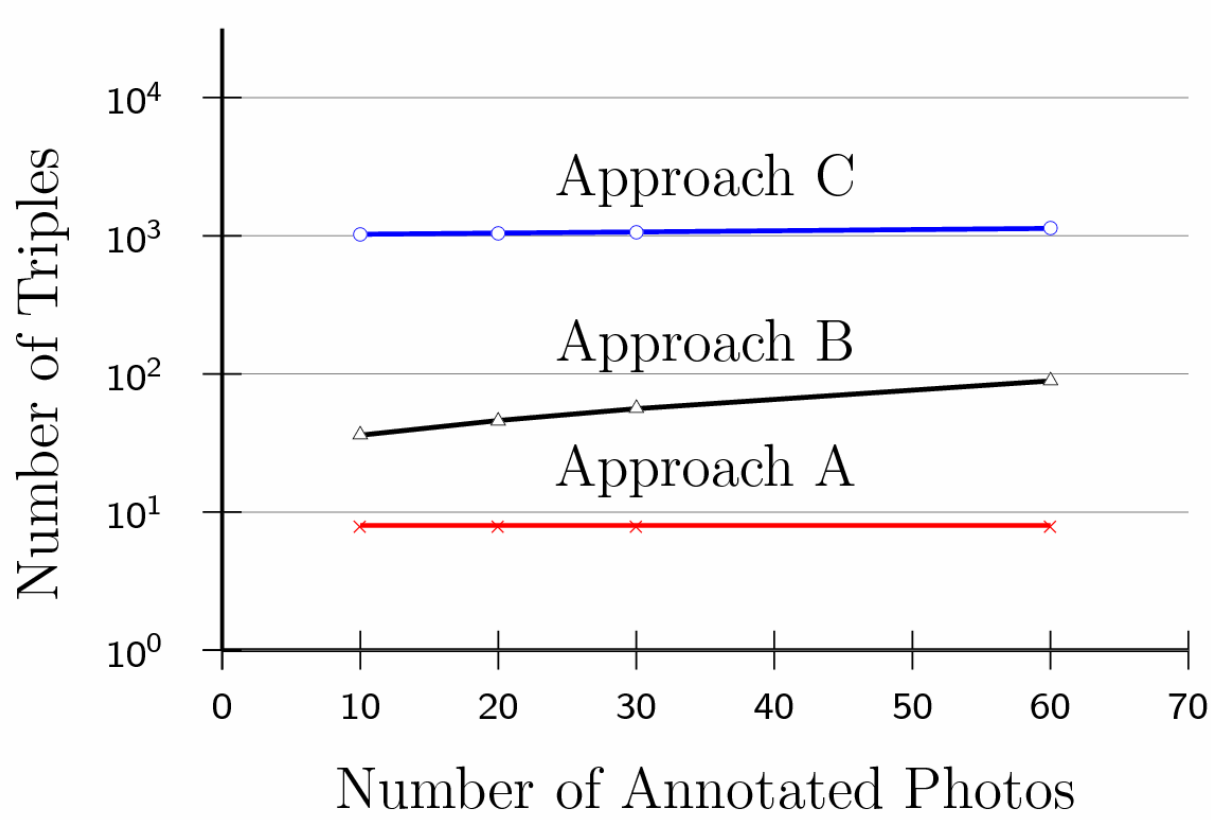
Showcase: PSIMeter

- **Approach A** uses the search functionality of the flickr API in a first step to retrieve the IDs of photos tagged with certain machine tags. In a second step the flickr API is used to retrieve the available metadata for each photo. Finally the result of the two previous steps is converted into an RDF representation, locally;
- **Approach B** uses the flickr API to retrieve all public photos firstly. It then uses a local XSL transformation to generate the RDF graph;
- **Approach C** retrieves all public photos, as in the Approach B. Then, for each photo an external service is invoked to generate the RDF graph.

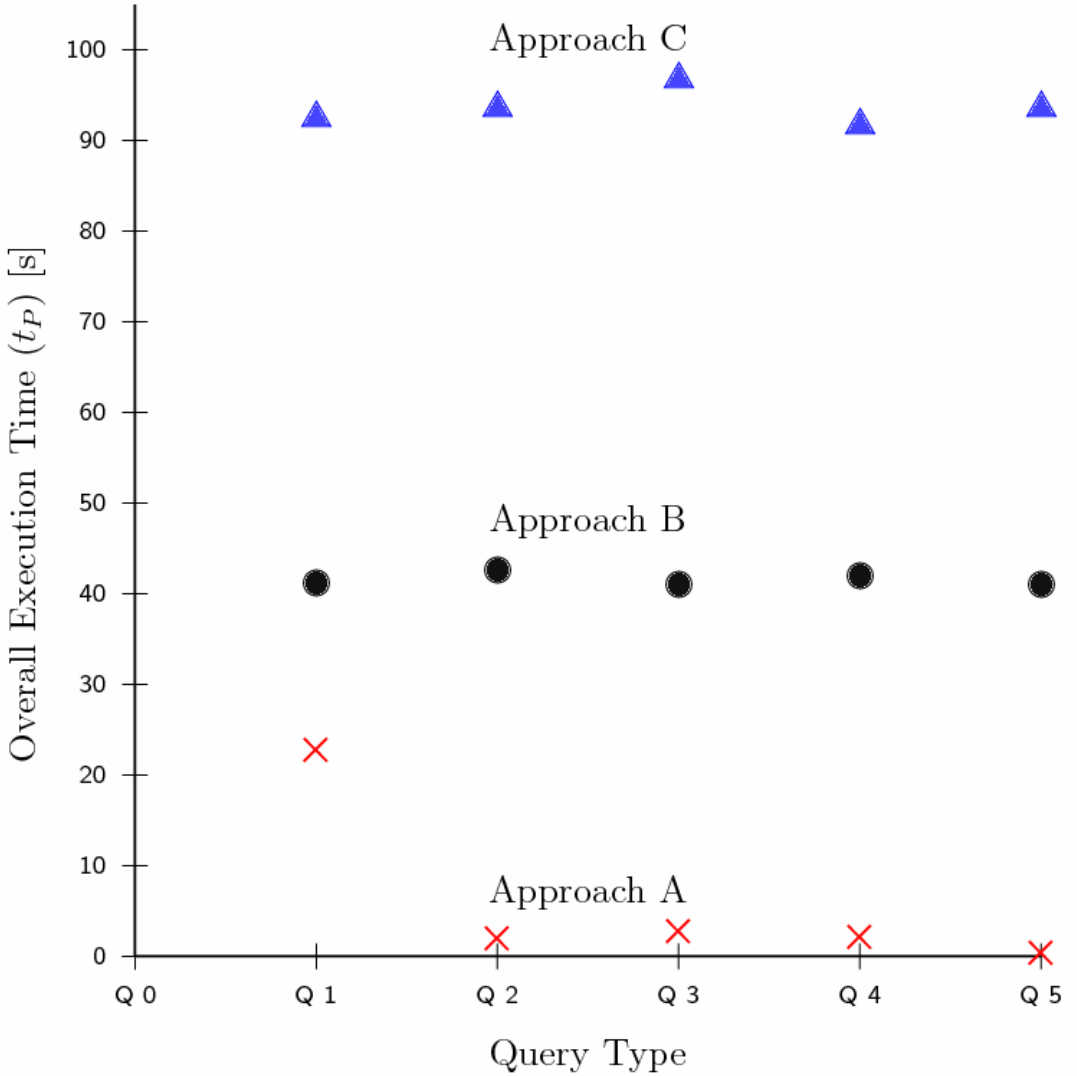
Showcase: PSIMeter



Showcase: PSIMeter



Showcase: PSIMeter



Conclusions

- RDF sources must be converted into in-memory representation to be processed; the selection of the type of sources is crucial (existing search infrastructure)
- Avoid conversion cascades (cf. hGRDDL)
- Achieving typical ‘Web response times’ (e.g. < 10s) is challenging, though doable

Discussion

- Now, let us discuss …