

Proceedings Of The First International Workshop On Consuming Linked Data
Shanghai, China, November 8, 2010

Silk Server

Adding missing Links while consuming Linked Data

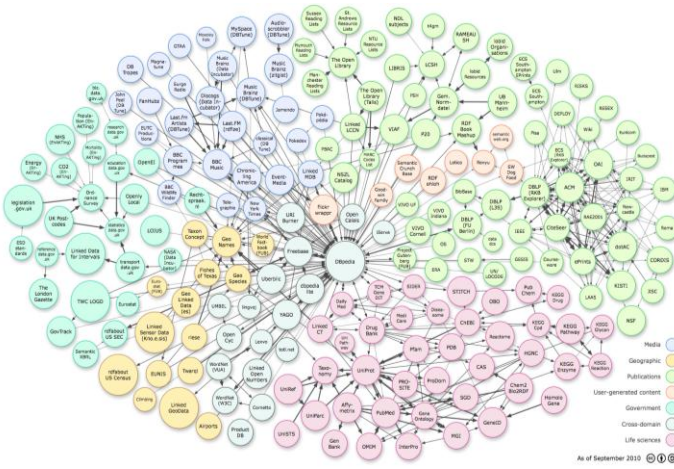
Robert Isele , Freie Universität Berlin

Anja Jentsch, Freie Universität Berlin

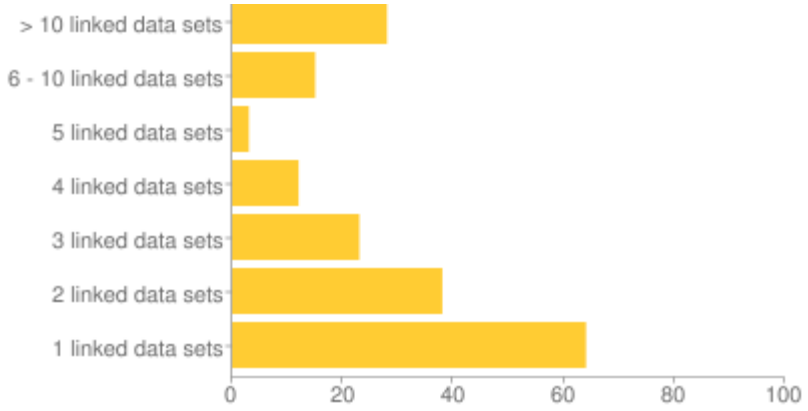
Christian Bizer, Freie Universität Berlin

A bad thing about Links

- The Semantic Web is a single global data space because data sources are connected by Links
- Many links are missing
- Setting links is effort for data publishers
- Tools enable data publishers to set links



Number of linked datasets	Number of datasets
1	64 (31.53 %)
2	38 (18.72 %)
3	23 (11.33 %)
4	12 (5.91 %)
5	3 (1.48 %)
6 to 10	15 (7.39 %)
more than 10	28 (13.79 %)



- 1. Silk Link Discovery Framework**
- 2. Silk Server**
- 3. Example Use Case**

The Silk Link Discovery Framework

- **Open source tool for discovering relationships between entities within different Linked Data sources.**

- **Main Features**

- 1. Open source link discovery framework, running on all major platforms**
- 2. Flexible, declarative language for specifying link conditions**
- 3. Works in situations where terms from different schemata are mixed**
- 4. Scalability and high performance through efficient data handling**
 - **Reduction of network load by caching and reusing of SPARQL result sets**
 - **Multi-threaded computation of the data item comparisons**
 - **Optional blocking of data items**

Silk Versions

■ Silk Single Machine

- Generate links on a single machine
- Local or remote data sets

■ Silk MapReduce

- Generate RDF links using a cluster of multiple machines
- Based on Hadoop (Can be run on Amazon Elastic MapReduce)

■ Silk Server

- Provides an HTTP API for matching instances from an incoming stream of RDF data while keeping track of known entities
- Can be used as an identity resolution component within applications that consume Linked Data from the Web
- Can be used for instance together with a Linked Data crawler to populate a local duplicate-free cache with data from the Web

Link Conditions

- Specify which conditions two entities must fulfill in order to be interlinked.
- A Link Condition is expressed as a combination of:

RDF paths

- Similar to SPARQL 1.1 Property Paths
- Examples:
 - ?movie/dbpedia:director/rdfs:label
 - ?person/label[@lang='en']

Transformations

- Transforms the result set of an RDF paths
- Variety of built-in transformations
- Examples:
 - LowerCase
 - RegexReplace
 - Stem

Similarity Metrics

- Similarity of two inputs based on a user-defined metric.
- Examples:
 - Various string similarity metrics
 - Geographic similarity
 - Date similarity

Aggregations

- Aggregates multiple similarity metrics
- Examples:
 - Min, Max, Average
 - Quadratic Mean
 - Geometric Mean

Example: Linking Persons

```
<LinkCondition>
  <Aggregate type="average">
    <Aggregate type="max" required="true">
      <Compare metric="jaroWinkler">
        <TransformInput function="lowerCase">
          <Input path="?a/foaf:name"/>
        </TransformInput>
        <TransformInput function="lowerCase">
          <Input path="?b/foaf:name"/>
        </TransformInput>
      </Compare>
    </Aggregate>
    <Aggregate type="max" weight="2" required="true">
      <Compare metric="levenshtein">
        <Input path="?a/foaf:homepage"/>
        <Input path="?b/foaf:homepage"/>
      </Compare>
      <Compare metric="equality">
        <Input path="?a/foaf:mbox_sha1sum"/>
        <Input path="?b/foaf:mbox_sha1sum"/>
      </Compare>
    </Aggregate>
  </Aggregate>
</LinkCondition>
```

Aggregate results

Compare names using JaroWinkler Similarity

Ignore character case

Compare homepages

Compare mailboxes

Linking Workflow

Blocking

- Partitions the instances into clusters. Only instances from the same cluster will be matched.
- This avoids matching the complete Cartesian product.

Matching

- Computes a similarity value for each pair of instances from the same cluster.
- The similarity value is based on a user-defined link condition.

Filtering

- Removes all links with a lower confidence than the user-defined threshold
- Only a limited number of links from the same subject are yielded

Performance Evaluation

- Finding links between cities in a dataset consisting of 10,500 settlements from DBpedia and 59,000 cities and towns from LinkedGeoData.

1. Without blocking (≈6 billion comparisons)

Silk Version	Link Generation Time	Number of Links
Silk Single Machine ¹	54 hours	9283
Silk MapReduce ²	6.7 hours	9283

2. With blocking (cities blocked by name using 50 blocks)

Silk Version	Link Generation Time	Number of Links
Silk Single Machine ¹	155.5 min	9224 (< 1 % loss)
Silk MapReduce ²	14.4 min	9224 (< 1 % loss)

1. Running on an Intel Core2Duo E8500 with 8GB of RAM
2. Running on Amazon Elastic MapReduce cluster consisting of 10 Amazon EC2 instances (High-CPU Medium Instance Profile)

Silk Server

- **Silk Single Machine and Silk MapReduce create links from static datasets.**
- **Silk Server generates links from an incoming stream of RDF data**

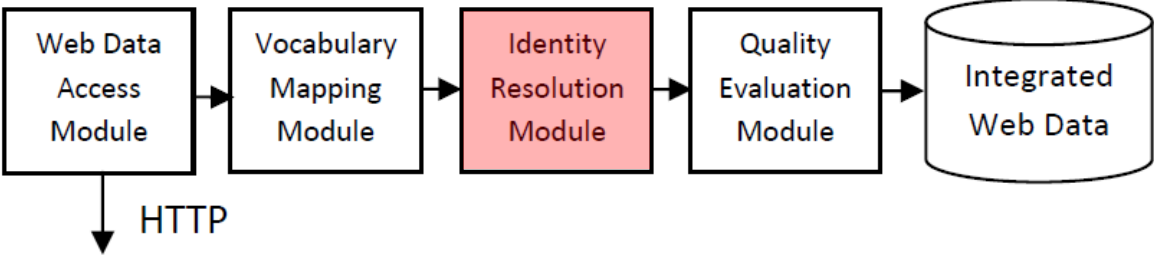
The big picture

Application Layer

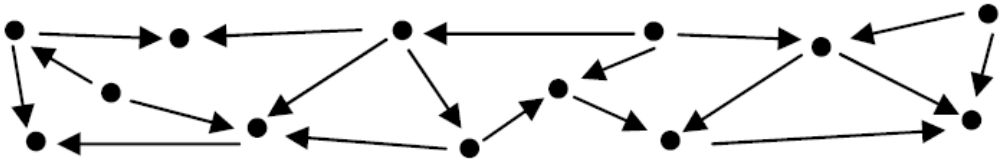


SPARQL

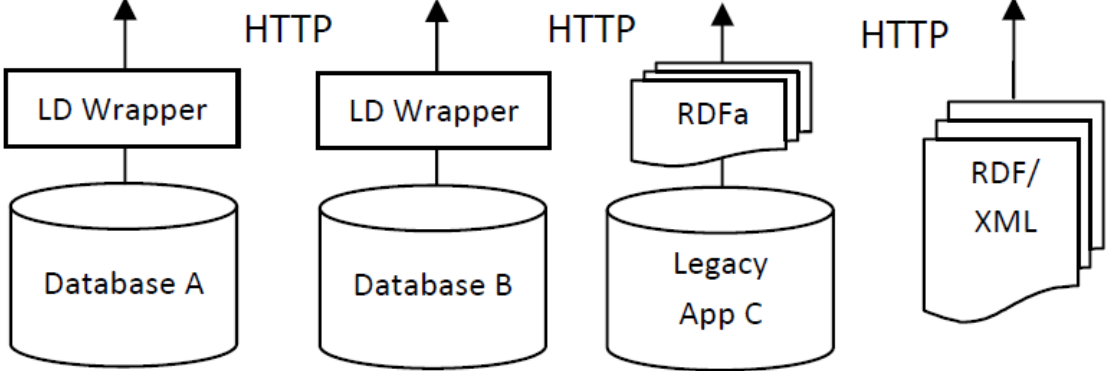
Data Access,
Integration and
Storage Layer



Web of Linked Data



Publication Layer



Silk Server Architecture

REST Interface

- Enables applications to commit newly discovered resources and receive the generated links
- Multiple requests can be processed in parallel

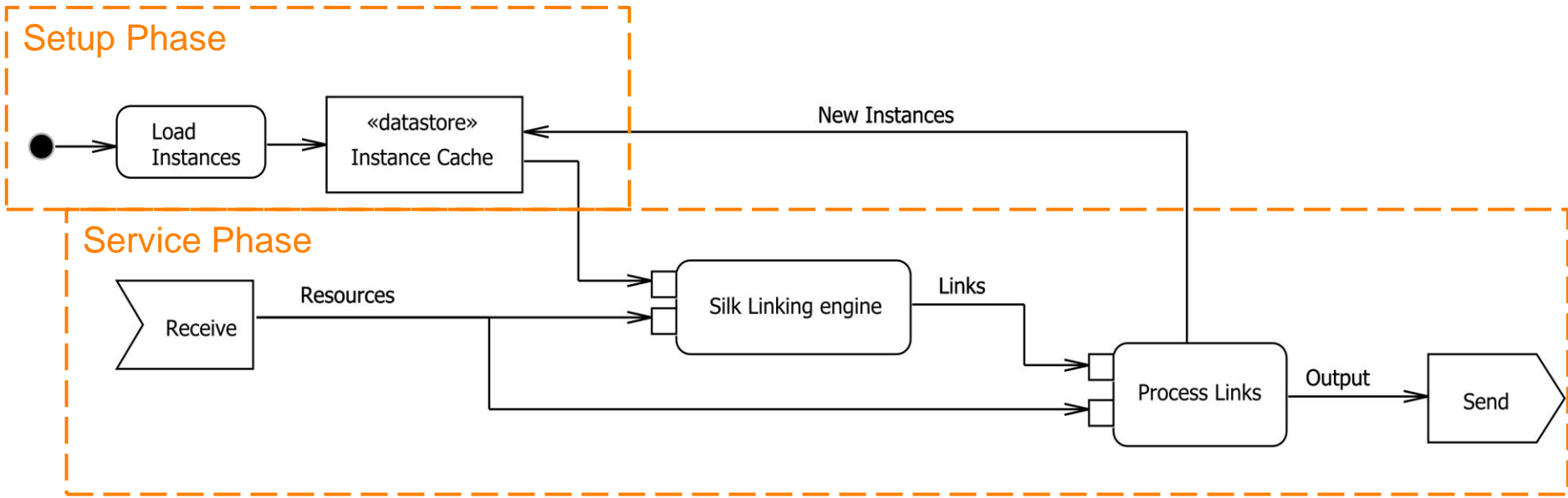
Silk Linking Engine

- Generates the links based on a set of link specifications

Instance Cache

- Holds all known instances and keeps track of newly discovered instances.
- Currently held in memory, but can be replaced by a persistent cache in future versions

Silk Server



Example: Semantic Web Conference Corpus

- Database of persons and papers from Semantic Web conferences
- For some persons, it contains links to the corresponding FOAF profile
- Many links are missing!
- Solution: LDSpider + Silk to get additional Links
 - To FOAF profiles
 - To Twitter accounts

Example Setup

- The previously shown link condition has been used to identify duplicate person descriptions

- The following steps have been executed:
 1. The Semantic Web Conference Corpus has been loaded into the Server
 2. LDSpider has been set up to crawl FOAF profiles
 3. LDSpider has been set up to crawl RDFa Twitter profiles

- All crawled documents are forwarded to Silk Server

Example Results

- In total, we have crawled 6730 FOAF profiles and 1160 Twitter accounts
- Silk Server identified the FOAF profiles of 228 persons
- Generated links have been evaluated
 - Sematic Web Conference Corpus links to 56 FOAF profiles
 - Silk Server reconstructed 51 profiles correctly
 - For some persons, Silk Server identified multiple profiles correctly
- Silk Server identified the Twitter profiles of 89 persons

Conclusion

- **Silk provides a Link Specification language which is expressive enough to cover all common use cases**
- **Silk provides a blocking feature as well as a MapReduce version to interlink big datasets.**
- **Silk Server matches instances from an incoming stream of RDF data and thus can be used as an identity resolution component within Linked Data applications**

Thanks!

Get Silk from: <http://www4.wiwiss.fu-berlin.de/bizer/silk>

This work was supported in part by Vulcan Inc. as part of its Project Halo (www.projecthalo.com) and by the EU FP7 project LOD2 - Creating Knowledge out of Interlinked Data (<http://lod2.eu/>, Ref. No. 257943).

