
DE: Department of Engineering Ferrara

# KRaider: a Crawler for Linked Data

Giuseppe Cota, Fabrizio Riguzzi, Riccardo Zese and Evelina Lamma

University of Ferrara

20 June 2019

# Table of contents

# Introduction

# Introduction

- The **Semantic Web** encourages the publication on the Internet of particular information, called semantic content, that can be processed and understood by machines.
- During the years, an increasing amount of data was published.
- Knowledge bases are represented with Semantic Web standards like RDF and OWL.
- In particular, knowledge bases in RDF are set of triples, i.e. statements of the form subject–predicate–object.
  - subject, predicate and object can be entities, i.e. they can denote something in the world (the "universe of discourse").
- **Linked data**: triples involving an entity can be distributed among different datasets.
- Due to its sheer size, it is difficult to explore or perform complex tasks such as inference on the whole Web of data.

DE: Department of Engineering Ferrara

# Introduction

- The **Semantic Web** encourages the publication on the Internet of particular information, called semantic content, that can be processed and understood by machines.
- During the years, an increasing amount of data was published.
- Knowledge bases are represented with Semantic Web standards like RDF and OWL.
- In particular, knowledge bases in RDF are set of triples, i.e. statements of the form subject–predicate–object.
  - subject, predicate and object can be entities, i.e. they can denote something in the world (the "universe of discourse").
- **Linked data**: triples involving an entity can be distributed among different datasets.
- Due to its sheer size, it is difficult to explore or perform complex tasks such as inference on the whole Web of data.

DE: Department of Engineering Ferrara

# Introduction

- The **Semantic Web** encourages the publication on the Internet of particular information, called semantic content, that can be processed and understood by machines.
- During the years, an increasing amount of data was published.
- Knowledge bases are represented with Semantic Web standards like RDF and OWL.
- In particular, knowledge bases in RDF are set of triples, i.e. statements of the form subject–predicate–object.
  - subject, predicate and object can be entities, i.e. they can denote something in the world (the "universe of discourse").
- **Linked data**: triples involving an entity can be distributed among different datasets.
- Due to its sheer size, it is difficult to explore or perform complex tasks such as inference on the whole Web of data.

DE: Department of Engineering Ferrara

# Introduction

- The **Semantic Web** encourages the publication on the Internet of particular information, called semantic content, that can be processed and understood by machines.
- During the years, an increasing amount of data was published.
- Knowledge bases are represented with Semantic Web standards like RDF and OWL.
- In particular, knowledge bases in RDF are set of triples, i.e. statements of the form subject–predicate–object.
  - subject, predicate and object can be entities, i.e. they can denote something in the world (the “universe of discourse”).
- **Linked data**: triples involving an entity can be distributed among different datasets.
- Due to its sheer size, it is difficult to explore or perform complex tasks such as inference on the whole Web of data.

DE: Department of Engineering Ferrara

# Introduction

- The **Semantic Web** encourages the publication on the Internet of particular information, called semantic content, that can be processed and understood by machines.
- During the years, an increasing amount of data was published.
- Knowledge bases are represented with Semantic Web standards like RDF and OWL.
- In particular, knowledge bases in RDF are set of triples, i.e. statements of the form subject–predicate–object.
  - subject, predicate and object can be entities, i.e. they can denote something in the world (the “universe of discourse”).
- **Linked data**: triples involving an entity can be distributed among different datasets.
- Due to its sheer size, it is difficult to explore or perform complex tasks such as inference on the whole Web of data.

# Introduction

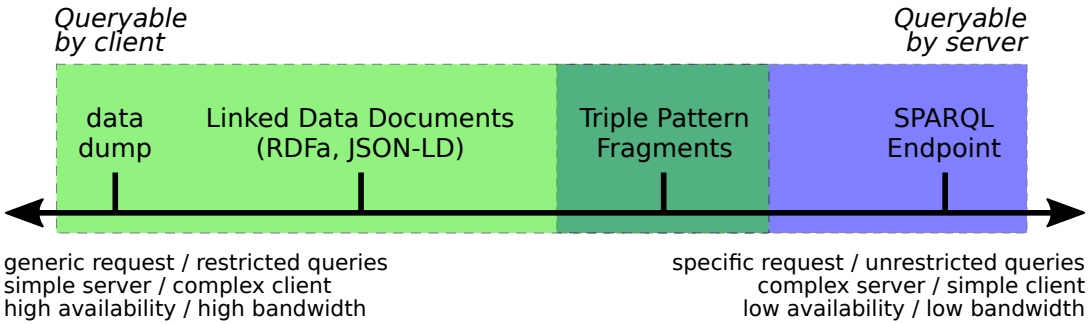
- The **Semantic Web** encourages the publication on the Internet of particular information, called semantic content, that can be processed and understood by machines.
- During the years, an increasing amount of data was published.
- Knowledge bases are represented with Semantic Web standards like RDF and OWL.
- In particular, knowledge bases in RDF are set of triples, i.e. statements of the form subject–predicate–object.
  - subject, predicate and object can be entities, i.e. they can denote something in the world (the “universe of discourse”).
- **Linked data**: triples involving an entity can be distributed among different datasets.
- Due to its sheer size, it is difficult to explore or perform complex tasks such as inference on the whole Web of data.

# Introduction

- The **Semantic Web** encourages the publication on the Internet of particular information, called semantic content, that can be processed and understood by machines.
- During the years, an increasing amount of data was published.
- Knowledge bases are represented with Semantic Web standards like RDF and OWL.
- In particular, knowledge bases in RDF are set of triples, i.e. statements of the form subject–predicate–object.
  - subject, predicate and object can be entities, i.e. they can denote something in the world (the “universe of discourse”).
- **Linked data**: triples involving an entity can be distributed among different datasets.
- Due to its sheer size, it is difficult to explore or perform complex tasks such as inference on the whole Web of data.

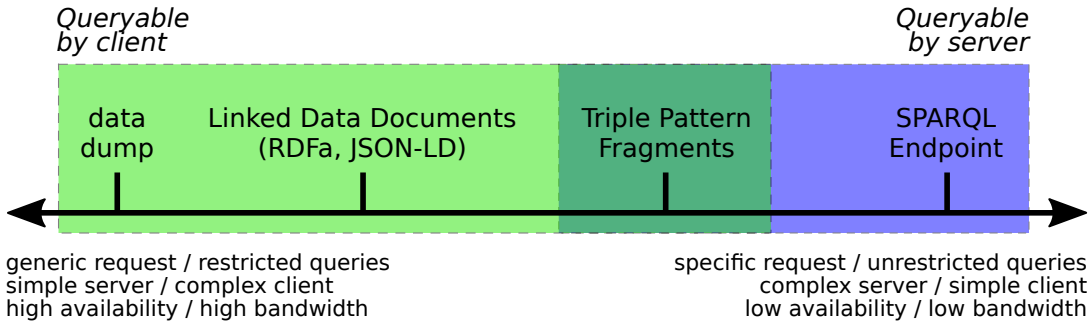
DE: Department of Engineering Ferrara

# Interfaces for querying Linked Data



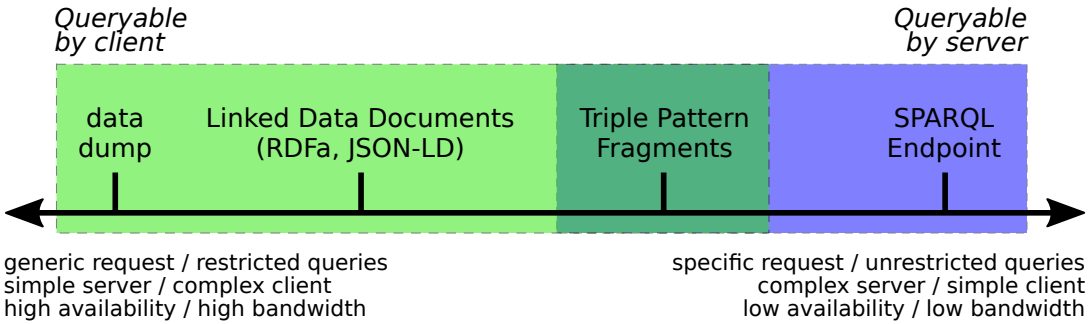

- **data dumps**: an archive (data dump) uploaded to a sever containing one or more files in an RDF syntax. The client downloads the data dump, extracts the contained files and processes them for performing queries.
- **Linked Data documents**: RDF triples are divided into several Linked Data documents organized by entity.
- **SPARQL Endpoints**: a client sends a (SPARQL) query to the server and delegates the entire execution to the server, which sends the answer (set of triples) back to the client.
- **Triple Pattern Fragments**: based on the SPARQL endpoint interface, it aims at reducing query execution costs by moving part of the execution workload from servers to clients.

# Interfaces for querying Linked Data



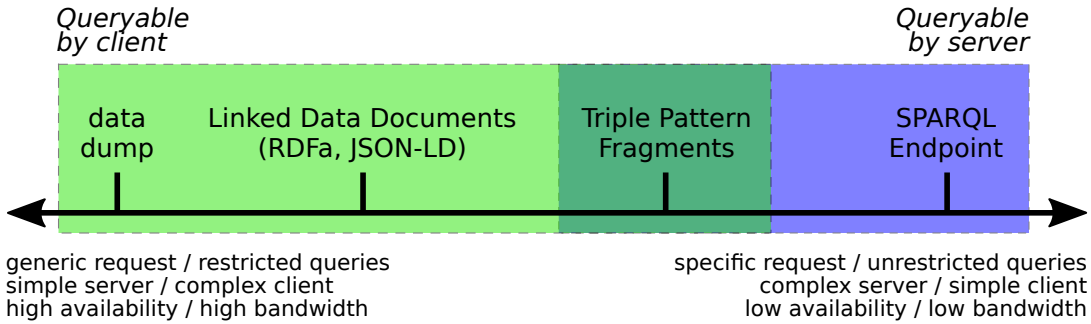

- **data dumps**: an archive (data dump) uploaded to a sever containing one or more files in an RDF syntax. The client downloads the data dump, extracts the contained files and processes them for performing queries.
- **Linked Data documents**: RDF triples are divided into several Linked Data documents organized by entity.
- **SPARQL Endpoints**: a client sends a (SPARQL) query to the server and delegates the entire execution to the server, which sends the answer (set of triples) back to the client.
- **Triple Pattern Fragments**: based on the SPARQL endpoint interface, it aims at reducing query execution costs by moving part of the execution workload from servers to clients.

# Interfaces for querying Linked Data




- **data dumps**: an archive (data dump) uploaded to a sever containing one or more files in an RDF syntax. The client downloads the data dump, extracts the contained files and processes them for performing queries.
- **Linked Data documents**: RDF triples are divided into several Linked Data documents organized by entity.
- **SPARQL Endpoints**: a client sends a (SPARQL) query to the server and delegates the entire execution to the server, which sends the answer (set of triples) back to the client.
- **Triple Pattern Fragments**: based on the SPARQL endpoint interface, it aims at reducing query execution costs by moving part of the execution workload from servers to clients.

# Interfaces for querying Linked Data




- **data dumps**: an archive (data dump) uploaded to a sever containing one or more files in an RDF syntax. The client downloads the data dump, extracts the contained files and processes them for performing queries.
- **Linked Data documents**: RDF triples are divided into several Linked Data documents organized by entity.
- **SPARQL Endpoints**: a client sends a (SPARQL) query to the server and delegates the entire execution to the server, which sends the answer (set of triples) back to the client.
- **Triple Pattern Fragments**: based on the SPARQL endpoint interface, it aims at reducing query execution costs by moving part of the execution workload from servers to clients.

# KRaider

# KRaider

**KRaider** ("Knowledge Raider"), is a system that retrieves the **relevant fragment of an entity** from different SPARQL endpoints, without the user knowing their location.

## Definition of Relevant Knowledge Fragment

The knowledge fragment $F_e$ relevant to an entity $e$ is the smallest fragment of the Web of data $\mathcal{W}$ ($F_e \subset \mathcal{W}$), such that a task involving entity $e$ and performed on $F_e$, provides the same results as if the task was performed on $\mathcal{W}$.

- In other words, we want to extract a fragment that holds enough information and that is small enough to allow the efficient execution of various tasks.

DE: Department of Engineering Ferrara

# KRaider

**KRaider** ("Knowledge Raider"), is a system that retrieves the **relevant fragment of an entity** from different SPARQL endpoints, without the user knowing their location.

## Definition of Relevant Knowledge Fragment

The knowledge fragment $F_e$ relevant to an entity $e$ is the smallest fragment of the Web of data $\mathcal{W}$ ($F_e \subset \mathcal{W}$), such that a task involving entity $e$ and performed on $F_e$, provides the same results as if the task was performed on $\mathcal{W}$.

- In other words, we want to extract a fragment that holds enough information and that is small enough to allow the efficient execution of various tasks.

# KRaider

**KRaider** ("Knowledge Raider"), is a system that retrieves the **relevant fragment of an entity** from different SPARQL endpoints, without the user knowing their location.

## Definition of Relevant Knowledge Fragment

The knowledge fragment $F_e$ relevant to an entity $e$ is the smallest fragment of the Web of data $\mathcal{W}$ ($F_e \subset \mathcal{W}$), such that a task involving entity $e$ and performed on $F_e$, provides the same results as if the task was performed on $\mathcal{W}$.

- In other words, we want to extract a fragment that holds enough information and that is small enough to allow the efficient execution of various tasks.

DE: Department of Engineering Ferrara

# Extraction of Knowledge Fragments

- The relevant fragment is extracted by recursively traversing the RDF graphs starting from an entity.
- Algorithm:
  1. Given an entity $e$ identified by an IRI $I_e$, it extracts the triples that have $e$ as subject, i.e. triples of the form $(e, p, o)$.
  2. The recursion depth is decremented and the objects $o$ of the obtained triples are used to extract additional knowledge until the user-defined recursion depth is reached.
  3. In case $p$ is equal to `owl:sameAs`, `owl:equivalentClass` or `owl:equivalentProperty`, the recursion depth is not decremented.
  4. If the object's IRI $I_o$ has a domain $D_o$ which is different from the domain of subject's IRI, in the next recursive step, the SPARQL endpoint hosted by $D_o$ is also queried.

# Extraction of Knowledge Fragments

- The relevant fragment is extracted by recursively traversing the RDF graphs starting from an entity.
- Algorithm:
  1. Given an entity $e$ identified by an IRI $I_e$, it extracts the triples that have $e$ as subject, i.e. triples of the form ($e$, $p$, $o$).
  2. The recursion depth is decremented and the objects $o$ of the obtained triples are used to extract additional knowledge until the user-defined recursion depth is reached.
  3. In case $p$ is equal to `owl:sameAs`, `owl:equivalentClass` or `owl:equivalentProperty`, the recursion depth is not decremented.
  4. If the object's IRI $I_o$ has a domain $D_o$ which is different from the domain of subject's IRI, in the next recursive step, the SPARQL endpoint hosted by $D_o$ is also queried.

# Extraction of Knowledge Fragments

- The relevant fragment is extracted by recursively traversing the RDF graphs starting from an entity.
- Algorithm:
  1. Given an entity *e* identified by an IRI $I_e$, it extracts the triples that have *e* as subject, i.e. triples of the form (*e*, *p*, *o*).
  2. The recursion depth is decremented and the objects *o* of the obtained triples are used to extract additional knowledge until the user-defined recursion depth is reached.
  3. In case *p* is equal to `owl:sameAs`, `owl:equivalentClass` or `owl:equivalentProperty`, the recursion depth is not decremented.
  4. If the object's IRI $I_o$ has a domain $D_o$ which is different from the domain of subject's IRI, in the next recursive step, the SPARQL endpoint hosted by $D_o$ is also queried.

# Extraction of Knowledge Fragments

- The relevant fragment is extracted by recursively traversing the RDF graphs starting from an entity.
- Algorithm:
  1. Given an entity $e$ identified by an IRI $I_e$, it extracts the triples that have $e$ as subject, i.e. triples of the form ($e$, $p$, $o$).
  2. The recursion depth is decremented and the objects $o$ of the obtained triples are used to extract additional knowledge until the user-defined recursion depth is reached.
  3. In case $p$ is equal to `owl:sameAs`, `owl:equivalentClass` or `owl:equivalentProperty`, the recursion depth is not decremented.
  4. If the object's IRI $I_o$ has a domain $D_o$ which is different from the domain of subject's IRI, in the next recursive step, the SPARQL endpoint hosted by $D_o$ is also queried.

DE: Department of Engineering Ferrara

# Extraction of Knowledge Fragments

- The relevant fragment is extracted by recursively traversing the RDF graphs starting from an entity.
- Algorithm:
  1. Given an entity $e$ identified by an IRI $I_e$, it extracts the triples that have $e$ as subject, i.e. triples of the form ($e$, $p$, $o$).
  2. The recursion depth is decremented and the objects $o$ of the obtained triples are used to extract additional knowledge until the user-defined recursion depth is reached.
  3. In case $p$ is equal to `owl:sameAs`, `owl:equivalentClass` or `owl:equivalentProperty`, the recursion depth is not decremented.
  4. If the object's IRI $I_o$ has a domain $D_o$ which is different from the domain of subject's IRI, in the next recursive step, the SPARQL endpoint hosted by $D_o$ is also queried.

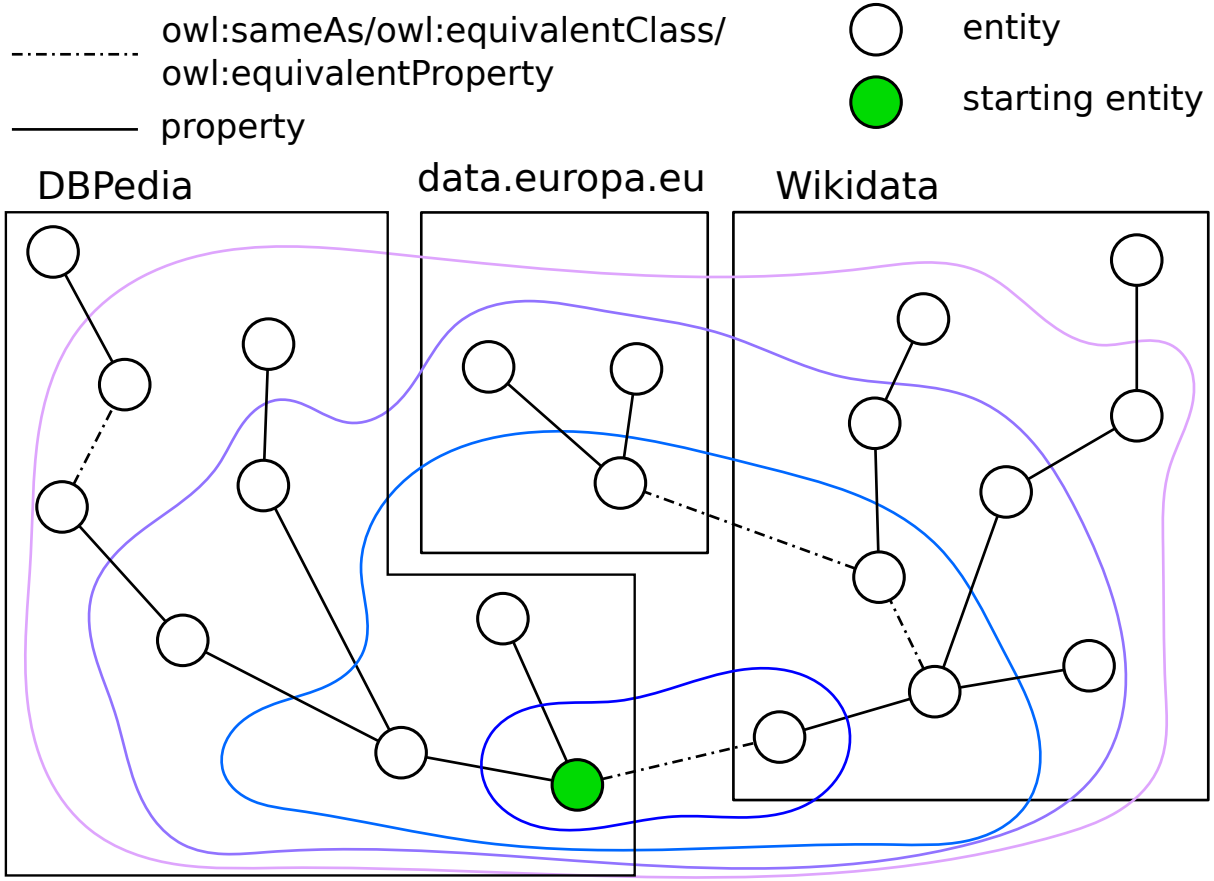
# Extraction of Knowledge Fragments

- The relevant fragment is extracted by recursively traversing the RDF graphs starting from an entity.
- Algorithm:
  1. Given an entity $e$ identified by an IRI $I_e$, it extracts the triples that have $e$ as subject, i.e. triples of the form ($e$, $p$, $o$).
  2. The recursion depth is decremented and the objects $o$ of the obtained triples are used to extract additional knowledge until the user-defined recursion depth is reached.
  3. In case $p$ is equal to `owl:sameAs`, `owl:equivalentClass` or `owl:equivalentProperty`, the recursion depth is not decremented.
  4. If the object's IRI $I_o$ has a domain $D_o$ which is different from the domain of subject's IRI, in the next recursive step, the SPARQL endpoint hosted by $D_o$ is also queried.

DE: Department of Engineering Ferrara

# Extraction of Knowledge Fragments

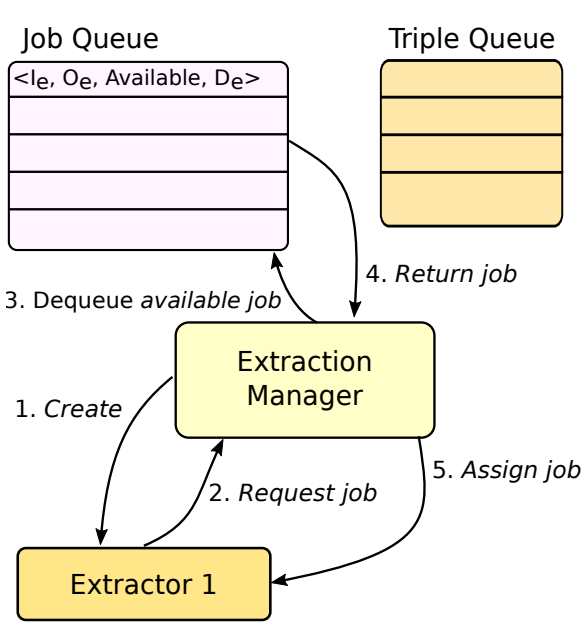
## Example



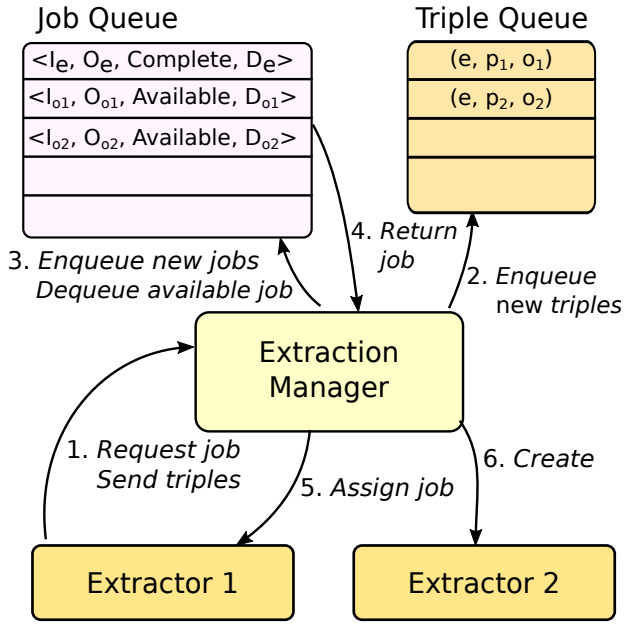

The smallest curve represent the fragment with recursion depth 0, the larger inner curve represents the fragment with recursion depth 1, up to the largest outer curve with recursion depth 3.

# KRaider's fragment extraction process

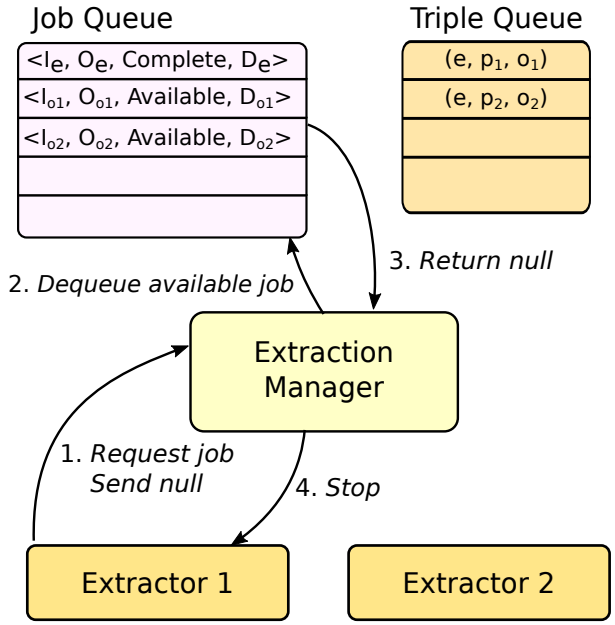
- An extractor is a component that *extracts* new triples from the Web of data.
- When an extractor is created, it is assigned to a new thread by a component called EXTRACTIONMANAGER, which manages the created extractors.
- EXTRACTIONMANAGER is responsible for the creation of new extractors, handles the extracted triples and assigns jobs to extractors.




(a)




(b)

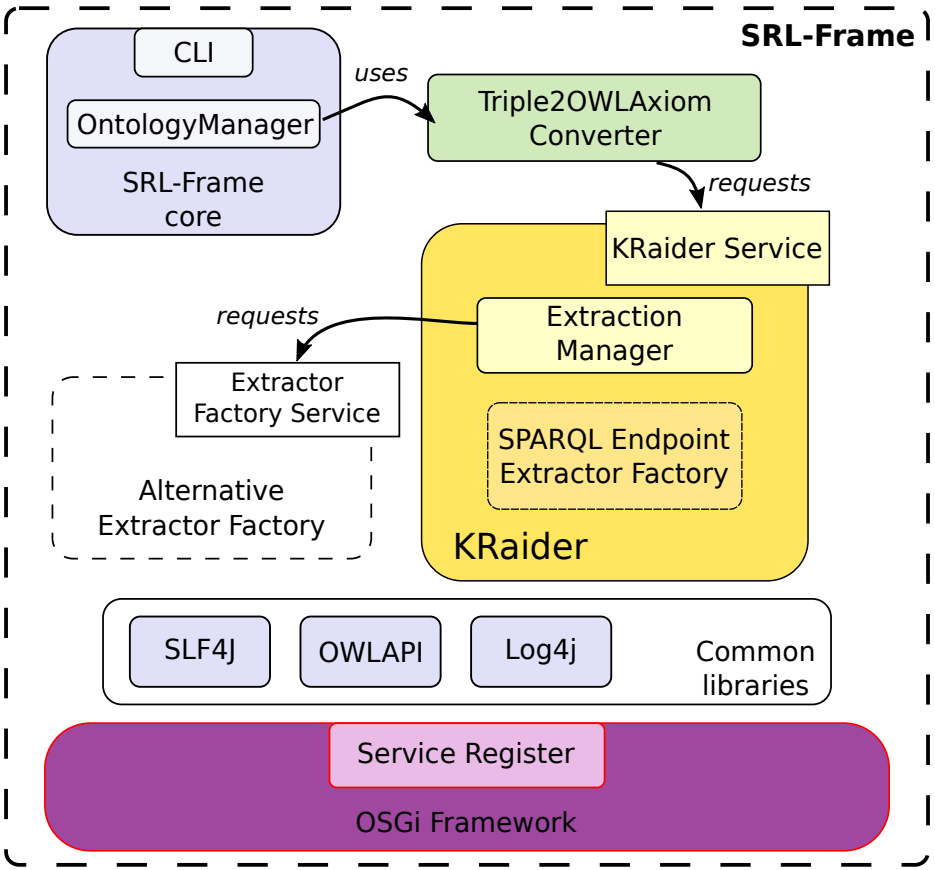



(c)

# SRL-Frame

# SRL-Frame architecture

- KRaider is integrated into SRL-Frame
- **SRL-Frame** ("Statistical Relational Learning Framework") is a framework (under development) written in Java and based on the OSGi technology.



# SRL-Frame architecture II

- The ultimate goal of KRaider is to be able to use several types of extractors, each exploiting a different Linked Data interface.
- The service oriented philosophy of OSGi comes in handy to realize this. In fact, KRaider can check, at run-time, which extraction services are available and then choose the ones to use.
- **At the moment**, KRaider only contains a single type of extractor which is able to exploit SPARQL endpoints to obtain triples.

# SRL-Frame architecture II

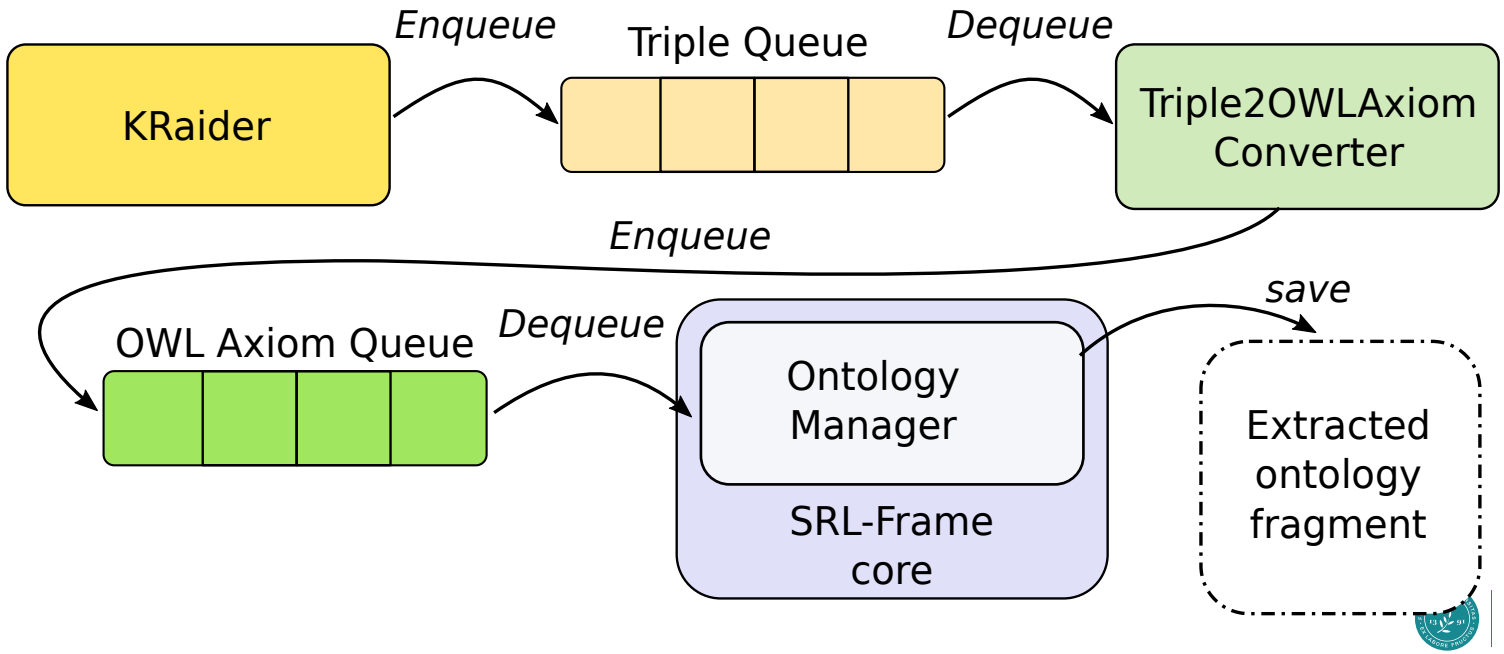
- The ultimate goal of KRaider is to be able to use several types of extractors, each exploiting a different Linked Data interface.
- The service oriented philosophy of OSGi comes in handy to realize this. In fact, KRaider can check, at run-time, which extraction services are available and then choose the ones to use.
- **At the moment**, KRaider only contains a single type of extractor which is able to exploit SPARQL endpoints to obtain triples.

# SRL-Frame architecture II

- The ultimate goal of KRaider is to be able to use several types of extractors, each exploiting a different Linked Data interface.
- The service oriented philosophy of OSGi comes in handy to realize this. In fact, KRaider can check, at run-time, which extraction services are available and then choose the ones to use.
- **At the moment**, KRaider only contains a single type of extractor which is able to exploit SPARQL endpoints to obtain triples.

# OWL Conversion of the Extracted Fragment

- Each extracted triple is added by EXTRACTORMANAGER to a triple queue (TRIPLEQUEUE).
- Many reasoners are able to perform inference on OWL ontologies.
- We developed a conversion pipeline which converts the extracted triples into an OWL ontology.




DE: Department of Engineering Ferrara

# Evaluation

# KRaider's Evaluation

- We evaluated KRaider used inside SRL-Frame by performing several knowledge fragment extraction tasks.
- The tests were performed on GNU/Linux machine equipped with Intel Core i7-5500U CPU @ 2.40GHz with 6 extractor threads.

| IRI | Recursion Depth 0 | | 1 | | 2 | |
|---|---|---|---|---|---|---|
| | # Axioms | Time | # Axioms | Time | # Axioms | Time |
| db:Leonardo_da_Vinci | 48 | 55.268 | 2956 | 130.029 | 100520.6 | 1271.571 |
| db:Angela_Merkel | 41 | 47.306 | 2412.4 | 107.771 | 160465.8 | 721.854 |
| db:Nikola_Tesla | 43 | 47.628 | 2139 | 87.499 | 62700.8 | 680.563 |

# Conclusions

# Problems and Limitations

- KRaider cannot handle blank nodes: KRaider just ignores the triples that contain them.
  - These nodes are important in order to convert RDF triples into complex OWL class expressions or properties.
  - For instance, the OWL class expression $\exists hasChild.Person$ corresponds to the following RDF triples:

```
_:x rdf:type owl:Restriction .
_:x owl:onProperty hasChild .
_:x owl:someValuesFrom Person .
```

- After multiple executions, it could happen that triples about an entity were already been extracted.
  - In order to improve the performances, KRaider should perform some sort of caching.

# Problems and Limitations

- KRaider cannot handle blank nodes: KRaider just ignores the triples that contain them.
  - These nodes are important in order to convert RDF triples into complex OWL class expressions or properties.
  - For instance, the OWL class expression $\exists hasChild.Person$ corresponds to the following RDF triples:

```
_:x rdf:type owl:Restriction .
_:x owl:onProperty hasChild .
_:x owl:someValuesFrom Person .
```

- After multiple executions, it could happen that triples about an entity were already been extracted.
  - In order to improve the performances, KRaider should perform some sort of caching.

# Conclusions and Future Work

- KRaider is a tool for extracting the relevant fragment of a given entity (triples involving an entity) from different SPARQL endpoints.
- The extracted triples are then converted into OWL axioms by another component.
- KRaider is integrated into SRL-Frame.
- SRL-Frame is based on OSGi technologies, which allows the system to dynamically install and start new services, hence making the framework flexible to changes.
- **Future work:**
  - Handle blank nodes
  - Caching
  - Retrieve triples from Linked Data documents
  - Extraction of new triples from text
  - (Probabilistic) Reasoning
  - Web application for visualizing the graph of the extracted fragment

DE: Department of Engineering Ferrara

# Thank you for listening!

# Questions?

DE: Department of Engineering Ferrara