Improving e-commerce fraud investigations in virtual, inter-institutional teams

based on the Master Thesis by Andreas Gerlach

3 initial assumptions

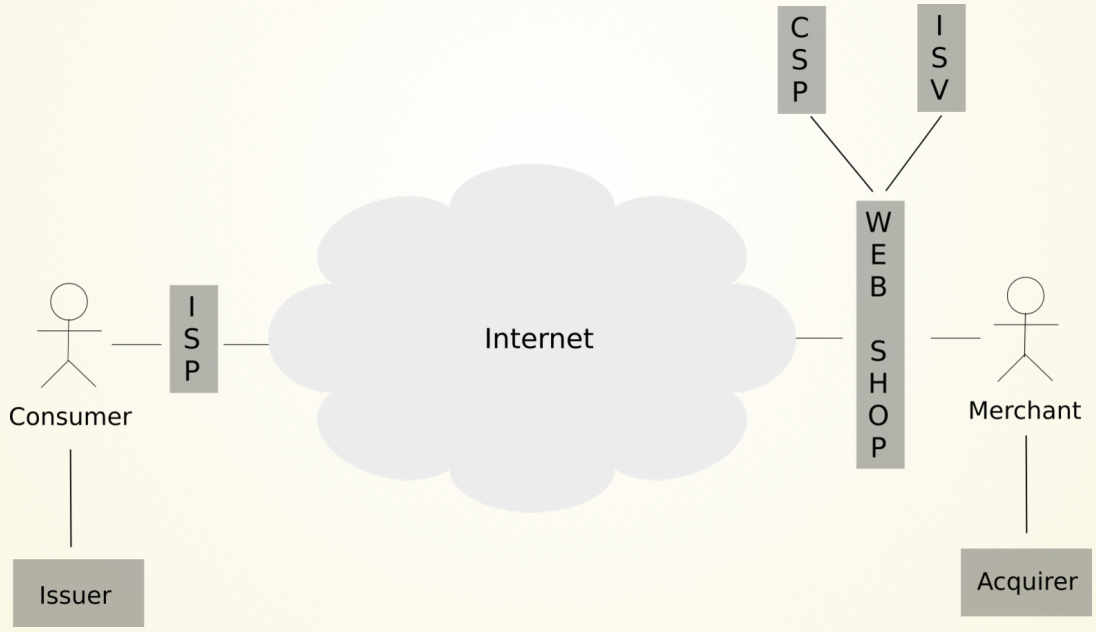
Transmission of credit card data over insecure co
mputer networks make them subjects to fraud.

Fraud prevention systems can cover up to 80% of f
raudulent transactions.

Investigation of edge cases involves elaborate ma
nual processes on the issuers.

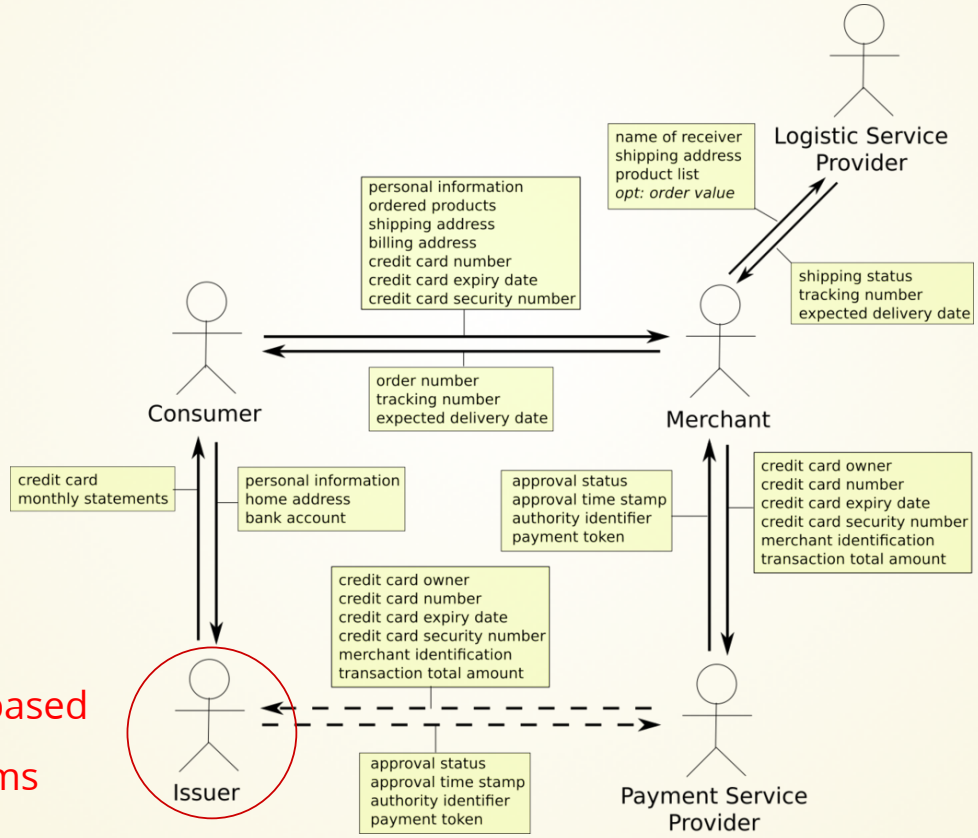
Transmission of credit card data over insecure co

mputer networks make them subjects to fraud.



Fraud prevention systems can cover up to 80% of f

raudulent transactions.



rule- or score-based

rating systems

Investigation of edge cases involves elaborate manual processes on the issuers:

- figure out contact details of each merchant affected,
- send an inquiry to each merchant asking for order details of recent purchases,
- collect, combine and analyse the information received,
- decide on edge cases based on consumer behaviour.

Hypothesis

Keeping the amount of fraudulent transactions low can not be solved by technology alone.

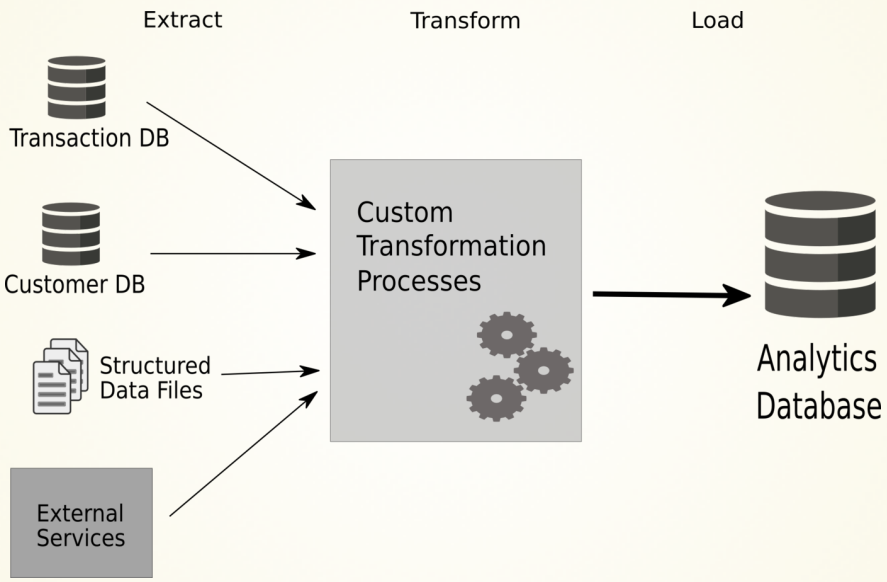
Edge cases are not evaluated in-depth, but rather be acknowledged after first plausibility checks.

A shared information space can bring together the know-how of relevant experts to improve the decision-making process.

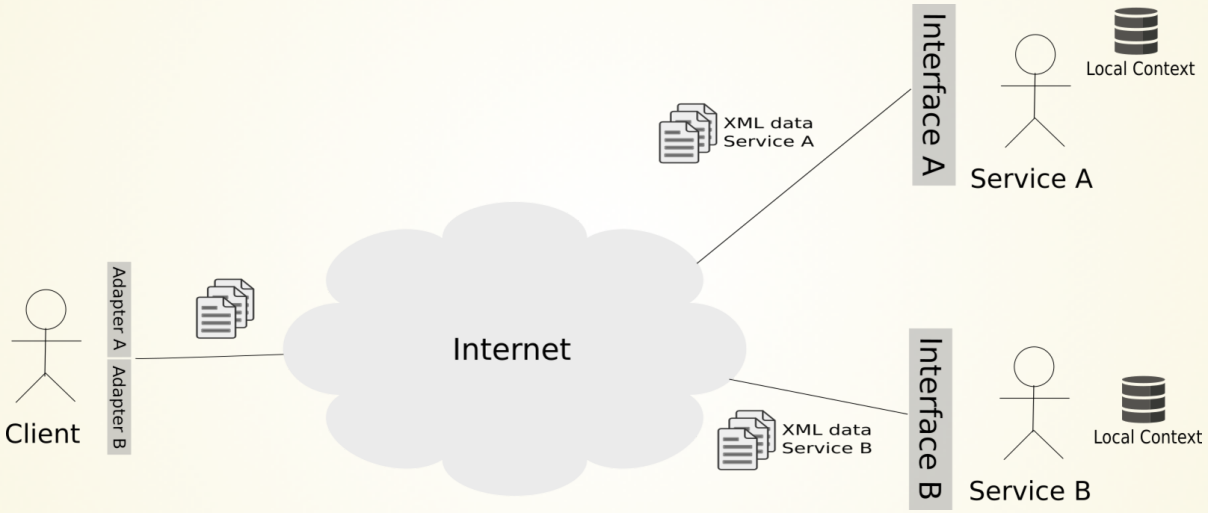
A collaborative system that combines information from relevant stakeholders can improve the current situation of investigating suspicious e-commerce activities.

Current approaches
for information sharing

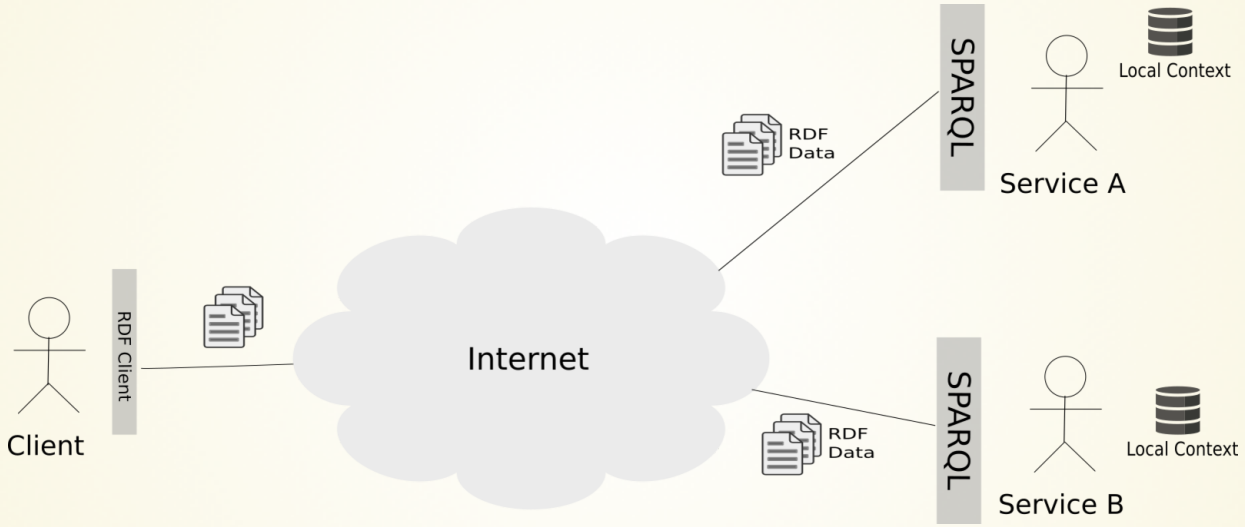
ETL


Extract
Transform
Load
Transaction DB
Customer DB
Structured
Data Files
External
Services
Custom
Transformation
Processes
Analytics
Database

Web Services

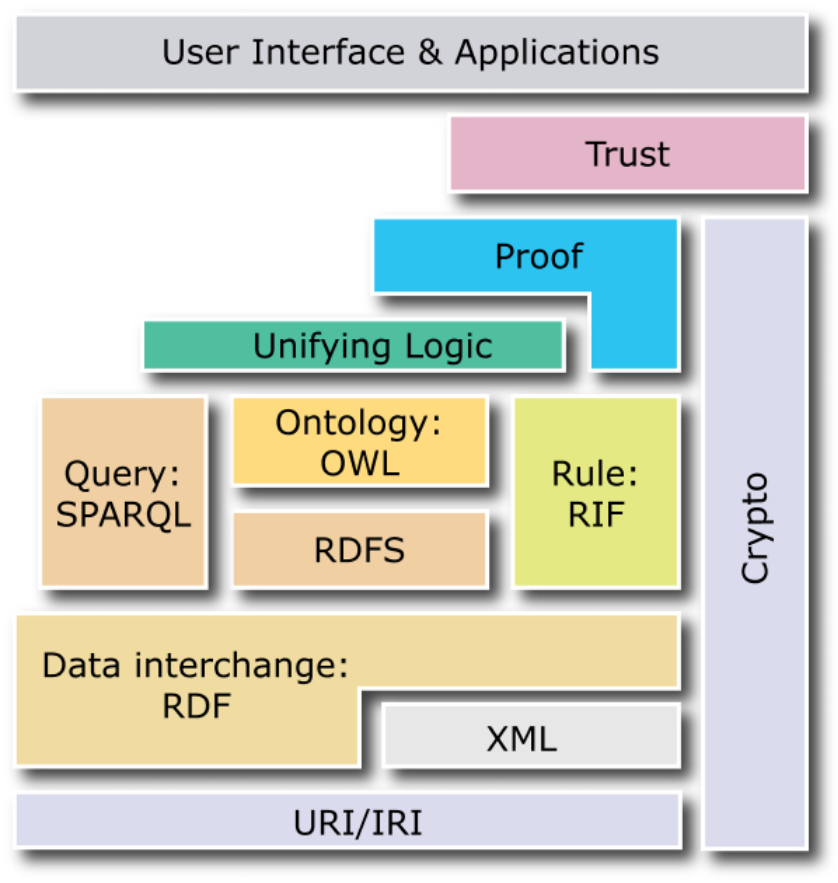


Semantic Web



Excursion: Semantic Web

Layered approach for data sharing on the Web



RDF: describing resources on the Web

simple, graph-oriented model based on "triples"

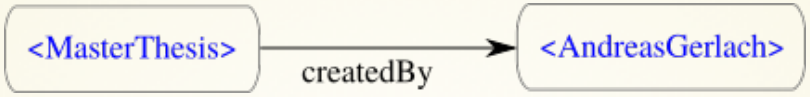
subject -- [predicate] --> object

using URI/URL to refer to objects and their predicates

different forms of serialization (interchangeable)

- **RDF/XML**: original XML-based approach
- **RDFa**: embed meta data into HTML
- **JSON-LD**: express triples in JSON
- **Turtle**: human-friendly, textual representation

RDF: describing resources on the Web




```
<?xml version="1.0"?>
<rdf:RDF xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
         xmlns:ex="http://www.example.com/">
  <rdf:Description rdf:about="http://www.example.com/MasterThesis">
      <ex:createdBy rdf:resource="http://www.example.com/AndreasGerlach"/>
  </rdf:Description>
</rdf:RDF>
```

```
<div about="http://www.example.com/MasterThesis">
  <span rel="http://www.example.com/createdBy"
        resource="http://www.example.com/AndreasGerlach">
</div>
```

```
{
  "@context": "http://www.example.com/",
  "@id": "http://www.example.com/MasterThesis",
  "createdBy": "http://www.example.com/AndreasGerlach"
}
```

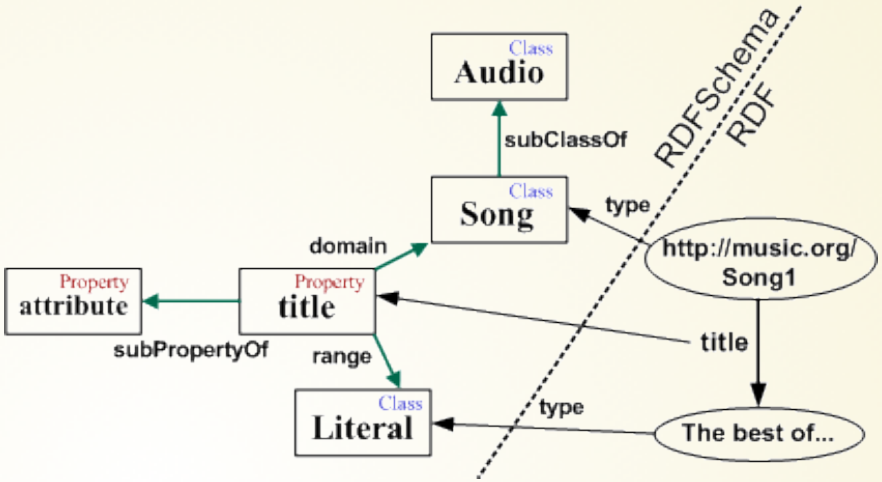
```
@prefix ex:<http://www.example.com/> .
ex:MasterThesis ex:createdBy ex:AndreasGerlach .
```

RDFS: of classes and their relationships

Classes	Used for
rdfs:Resource	individual resources
rdfs:Class	classes
rdfs:Literal	literals
rdfs:Propery	properties

Predicates	Describes
rdf:type	kind of class
rdfs:subClassOf	inheritance between classes
rdfs:subPropertyOf	inheritance between properties
rdfs:domain	restrict the subjects of a property
rdfs:range	restrict the values of a property

A small sample...




```
# define prefixes for URIs
@prefix ex: <http://www.example.com/> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .

# define available classes and their hierarchy
ex:Audio rdf:type rdfs:Class .
ex:Song  rdf:type rdfs:Class;
         rdfs:subClassOf ex:Audio .

# define available properties, their hierarchy and restrictions
ex:attribute rdf:type rdfs:Property .
ex:title rdf:type rdfs:Property;
         rdfs:subPropertyOf ex:attribute;
         rdfs:domain ex:Song;
         rdfs:range rdfs:Literal .

# add specific instances of classes
<http://music.org/Song1> rdf:type ex:Song;
                         ex:title "The best of..." .
```

SPARQL: an open protocol and query language

tabular output: *show all titles of songs in triple store!*

```
# define prefixes for URIs
PREFIX ex:  <http://www.example.com/> .
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .

# choose to output any title found in graph-pattern
SELECT ?title
WHERE {
  # describe the conditions for the query
  # as graph-patterns that have to match
  # here: it has to be a Song, which has a predicate title
  ?song rdf:type ex:Song .
  ?song ex:title ?title .
}
```

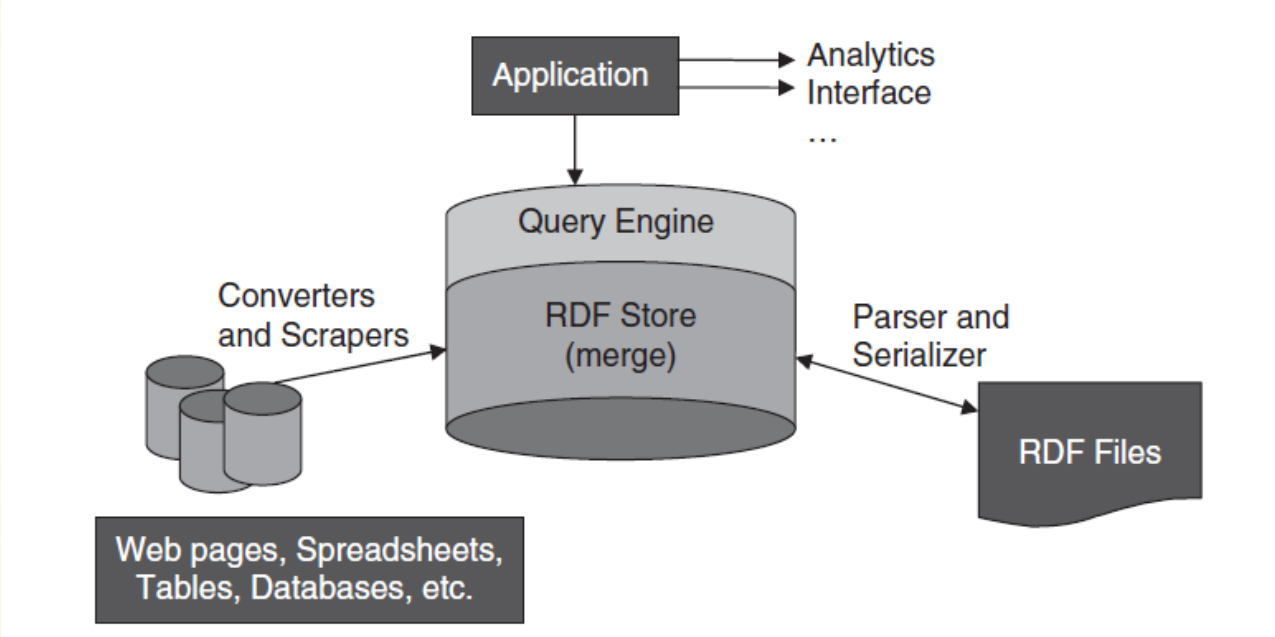
SPARQL: an open protocol and query language

graph patterns: *map song information to DublinCore!*

```
# define prefixes for URIs
PREFIX ex:  <http://www.example.com/> .
PREFIX dc:  <http://purl.org/dc/elements/1.1/> .
PREFIX dct: <http://purl.org/dc/terms/> .
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .

# create a new graph with the mapped song information
CONSTRUCT {
  ?song rdf:type dc:Sound .
  ?song dct:title ?title .
}
WHERE {
  # describe the conditions for the query
  # as graph-patterns that have to match
  # here: it has to be a Song, which has a predicate title
  ?song rdf:type ex:Song .
  ?song ex:title ?title .
}
```

Semantic Web application architecture



Conclusion

ETL is well suited for collecting and combining i nformation within a single company.

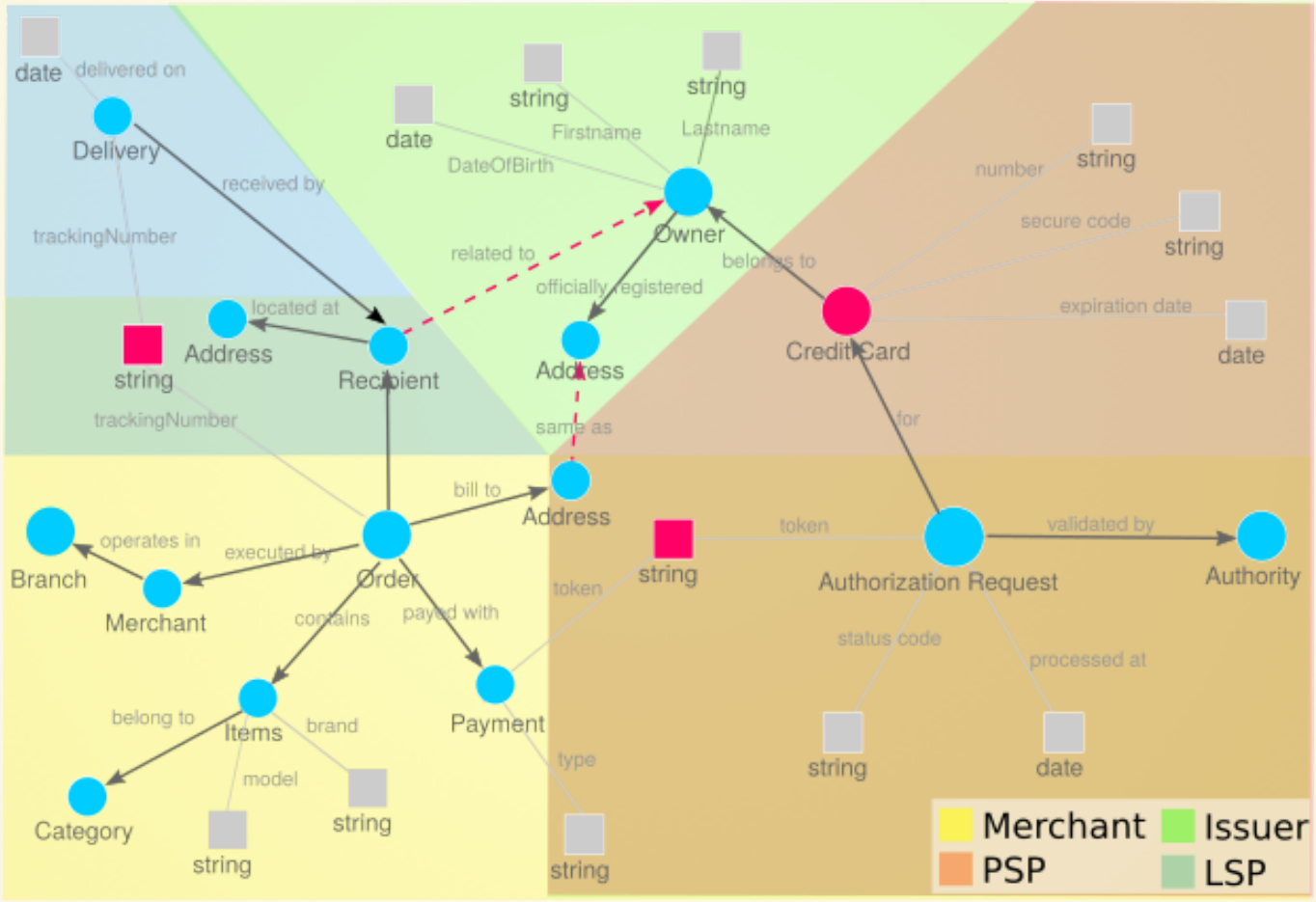
Web Services require adaptation to each service int erface that has to be integrated, do not provide th e semantics of the information exchanged.

Semantic Web do not restrict access to sensitive information, make any of them publicly available on the Web.

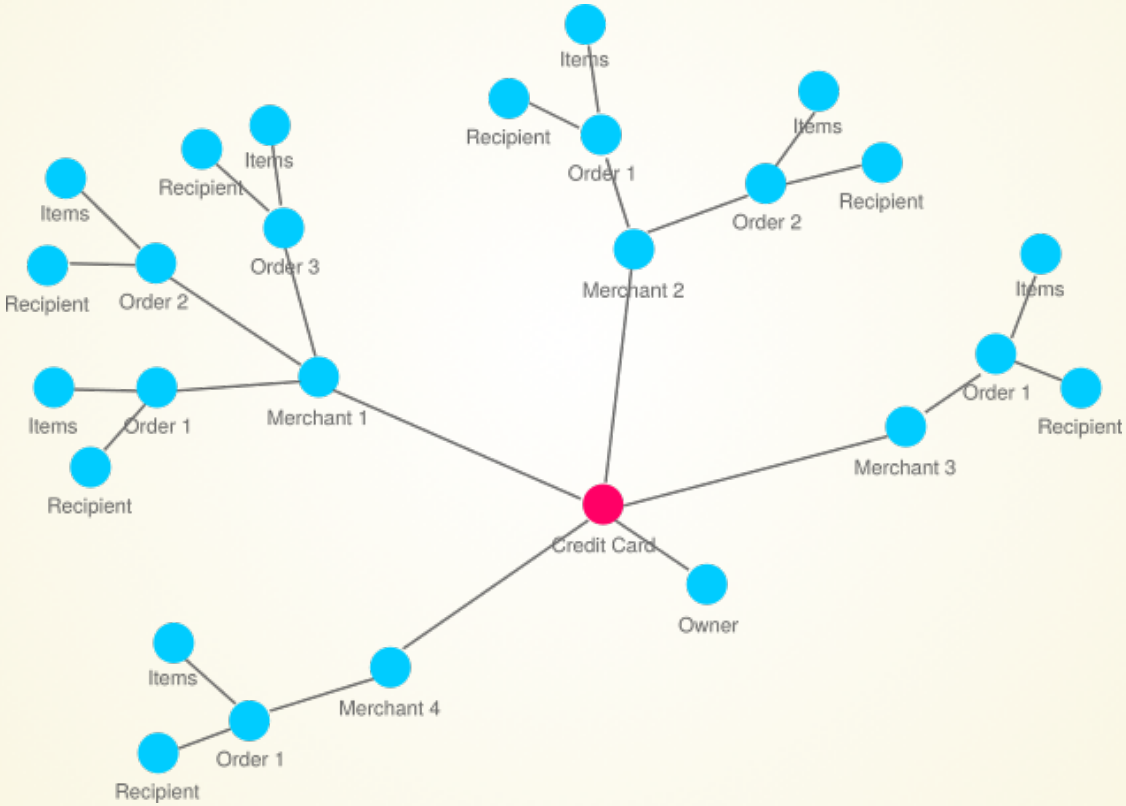
Proposal

The collaborative system should use fundamental technologies of the Semantic Web for describing resources as well as P2P technologies for securely sharing them.

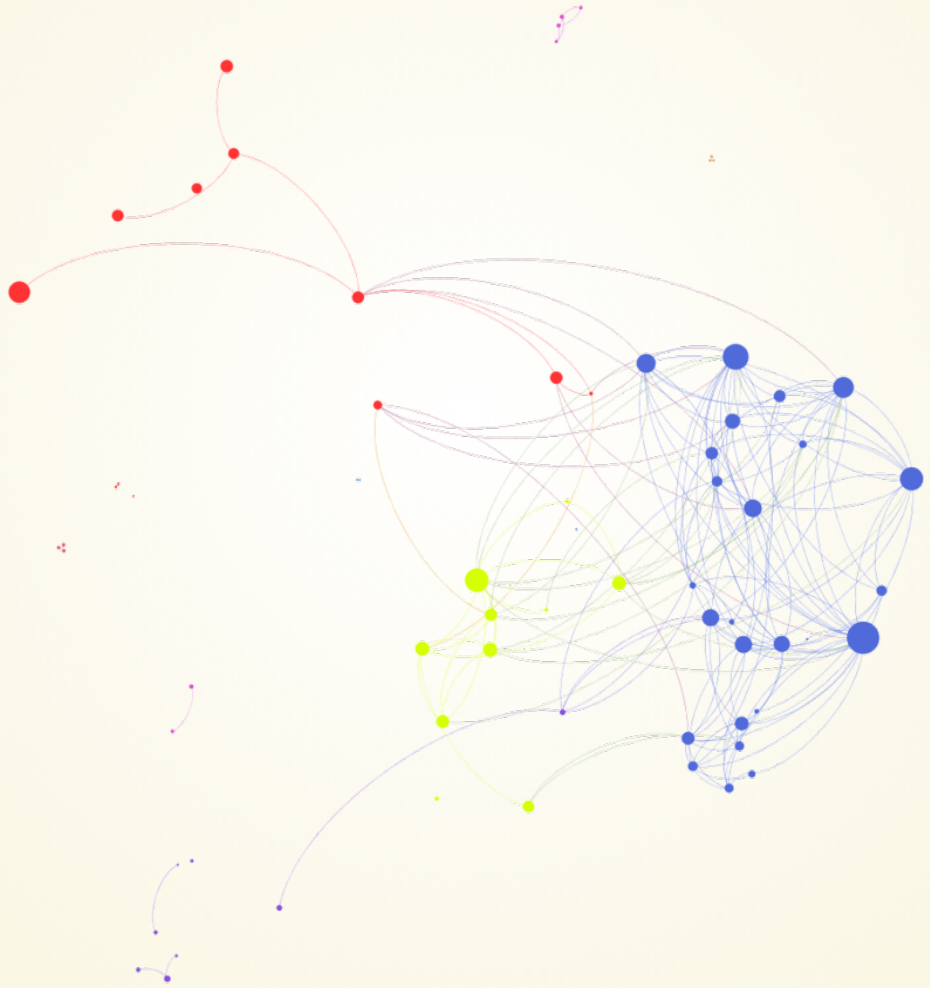
A combined data model of e-commerce activities



Initial collection of recent e-commerce activities



Classifying and clustering of e-commerce activities



RDF vocabularies and ontologies

- follow core principles of the Web

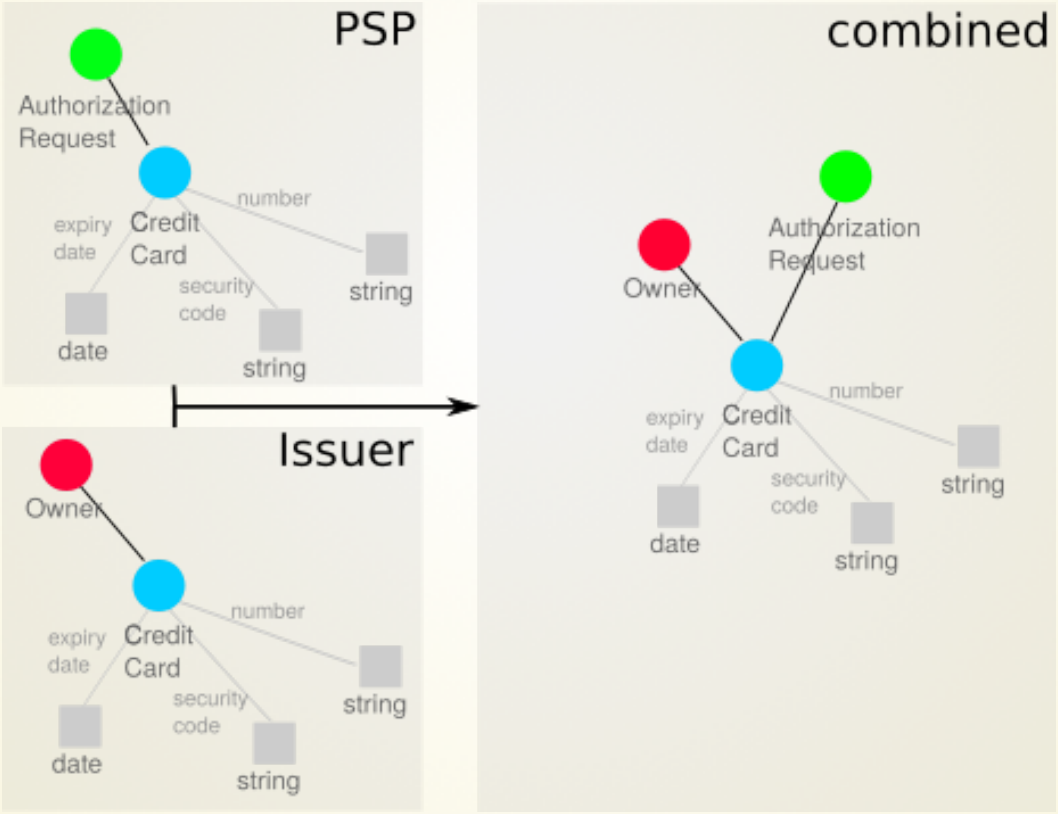
Mapping of RDF vocabularies

- classes and properties, equality or inheritance

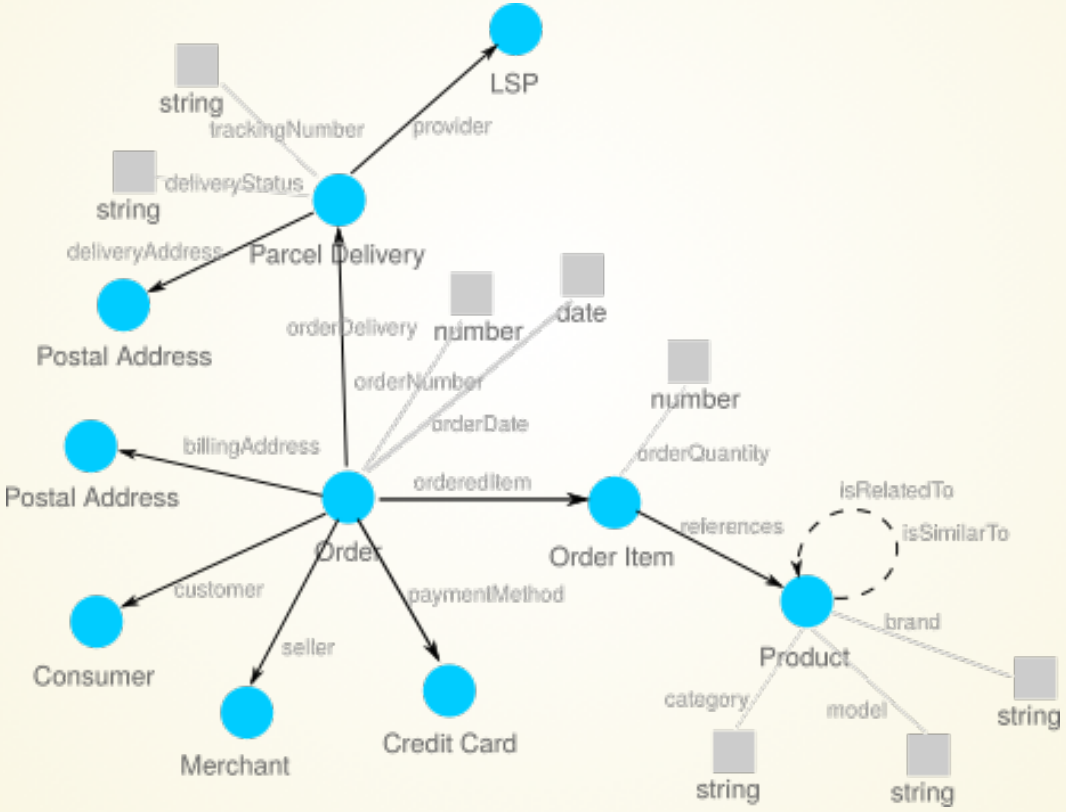
Linking of individual resources

- use unique identifiers or infer equality

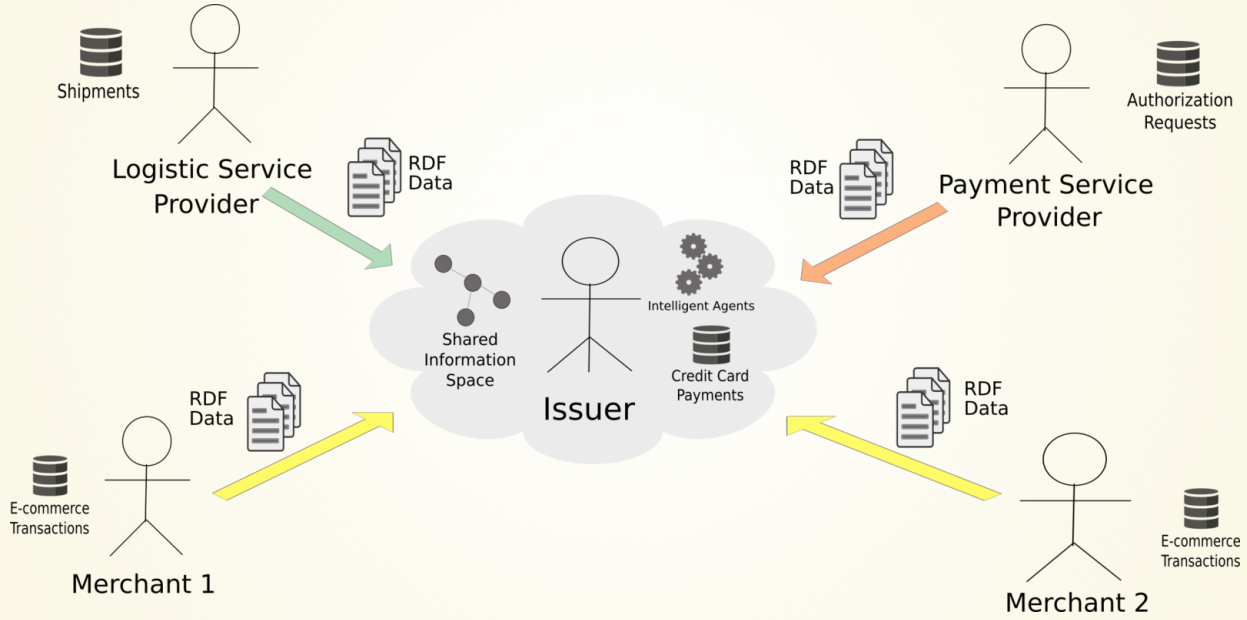
Build-in RDF merging capabilities



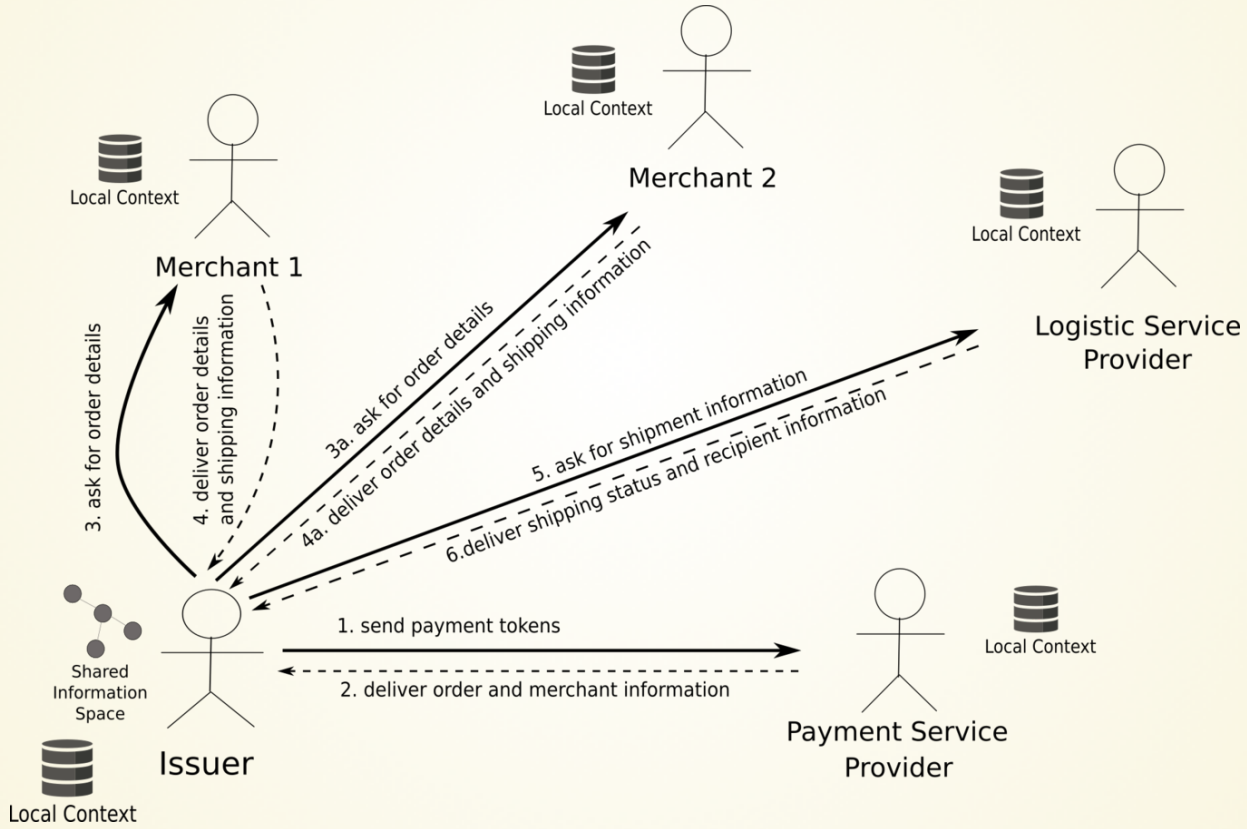
Schema.org: an extensible, open data vocabulary



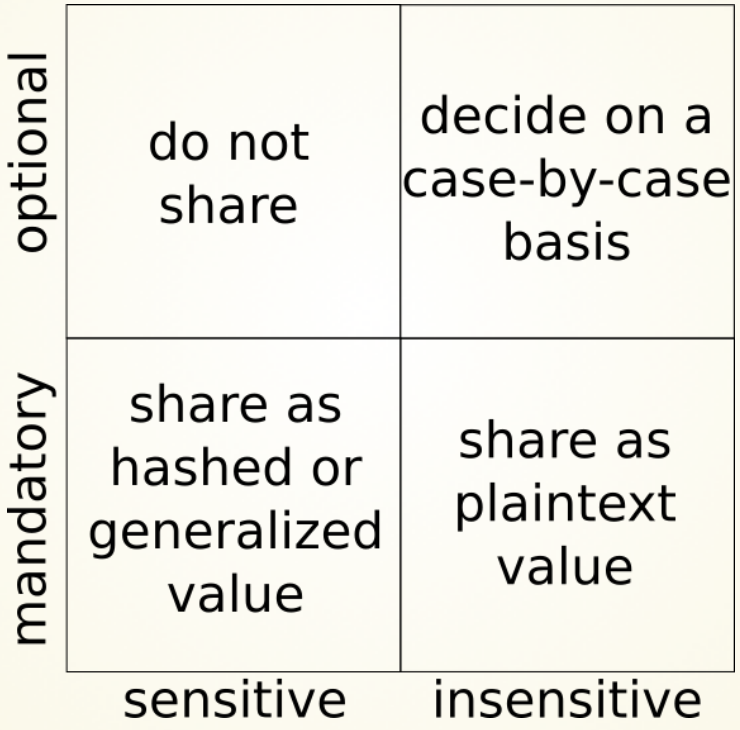
A partially centralized P2P collaborative system



Workflow


Local Context
Merchant 1
Local Context
Merchant 2
Local Context
Logistic Service Provider
3. ask for order details
4. deliver order details and shipping information
3a. ask for order details
4a. deliver order details and shipping information
5. ask for shipment information
6.deliver shipping status and recipient information
Shared Information Space
Issuer
Local Context
1. send payment tokens
2. deliver order and merchant information
Payment Service Provider
Local Context

Dealing with privacy concerns



Conclusion

Core Semantic Web technologies enable information sharing across organisational boundaries.

In combination with P2P technologies a secure collaborative system can be developed.

Graph-oriented representation of information, its visualization and clustering can support the investigation and knowledge generation process.

However, information have to be duplicated to the issuers as main actors (privacy concerns).

Different techniques to obfuscate sensitive information are available (hash, generalization).

A decentralized P2P system can solve most of the privacy issues. It needs further research though.

Thank you for your attention!