

# MatWare:

Constructing and Exploiting Domain Specific  
**Warehouses** by Aggregating **Semantic** Data

**Y. Tzitzikas**<sup>1,2</sup>, N. Minadakis<sup>1</sup>, Y. Marketakis<sup>1</sup>, P. Fafalios<sup>1,2</sup>,  
C. Alloca<sup>1</sup>, M. Mountantonakis<sup>1,2</sup>, I. Zidianaki<sup>1,2</sup>

<sup>1</sup> Institute of Computer Science, FORTH-ICS

<sup>2</sup> Computer Science Department, University of Crete, GREECE

# Outline

- Motivation
- Context & Requirements
- The Warehouse Construction Tool MatWare
  - The Process, Scope Control, Connectivity, Provenance, Architecture
- Applications of MatWare-constructed warehouses
- Concluding Remarks

# Motivation

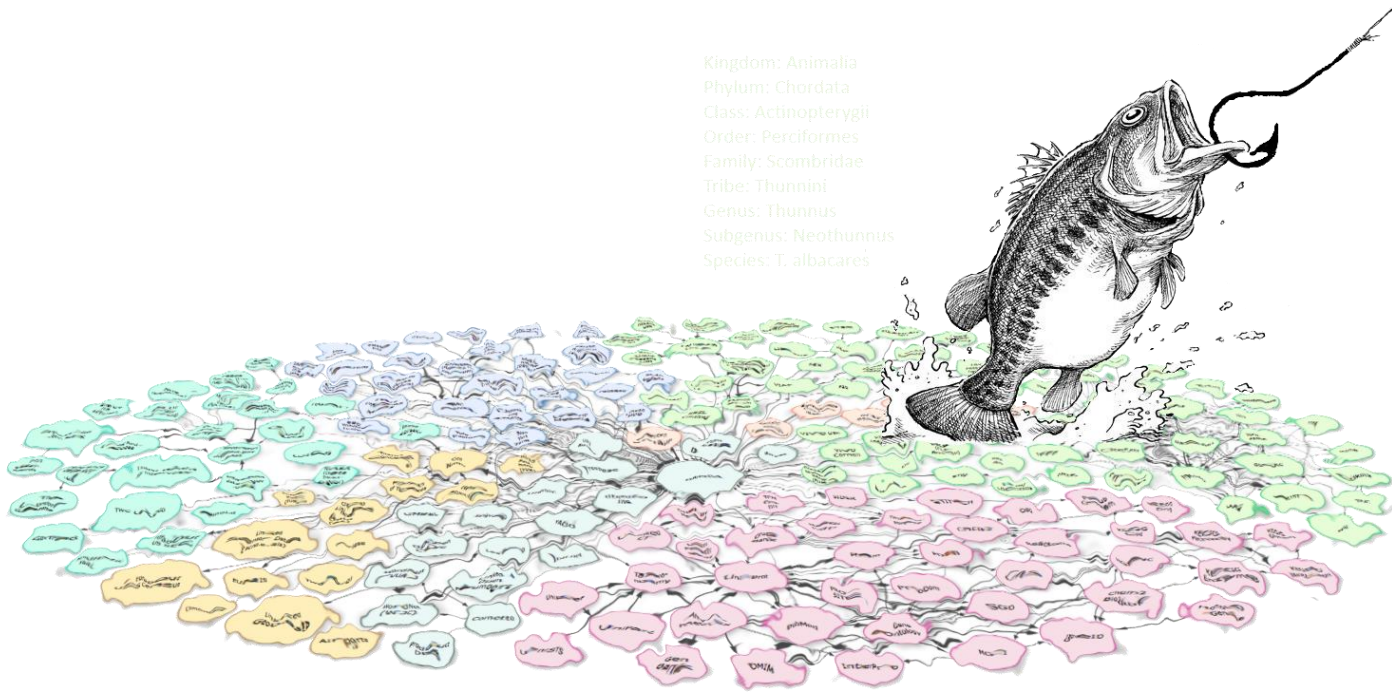
In many applications one has to **fetch** and **assemble pieces of information** coming from more than one sources (including SPARQL endpoints.)

*Def:* We use the term **Semantic Warehouse** (for short warehouse) to refer to a **read-only set of RDF triples fetched (and transformed) from different sources that aims at serving a particular set of query requirements.**

## Key Questions

- How to define the **objectives** and the **scope** of such a warehouse
- How to **connect** the **fetched pieces** of information?
- How to test that its **contents** meet the **objectives**?
- How to measure the **quality** of the warehouse?
- How to tackle the various issues of **provenance** that arise?
- How to **automate** the construction/maintenance process?

Kingdom: Animalia  
Phylum: Chordata  
Class: Actinopterygii  
Order: Perciformes  
Family: Scombridae  
Tribe: Thunnini  
Genus: Thunnus  
Subgenus: Neothunnus  
Species: T. albacares



# Context

# Context: iMarine



**Id:** It is an FP7 Research Infrastructure Project (2011-2014)

**Final goal:** launch an initiative aimed at establishing and operating an e-infrastructure supporting the principles of the Ecosystem Approach to fisheries management and conservation of marine living resources.

## Partners:



HELLENIC REPUBLIC

National and Kapodistrian  
University of Athens



Yannis Tzitzikas et al., ESWC 2014,  
Heraklion, Crete

# Marine Information: **in several sources**



**WoRMS:** World Register of Marine Species  
Registers more than 200K species



**ECOSCOPE-** A Knowledge Base About Marine Ecosystems (IRD, France)



**FLOD** (Fisheries Linked Data) of Food and Agriculture Organization (**FAO**) of the United Nations



**FishBase:** Probably the largest and most extensively accessed online database of fish species.



**DBpedia**

Yannis Tzitzikas et al., ESWC 2014,  
Heraklion, Crete

# Marine Information: in several sources

Storing  
**complementary  
information**



Taxonomic information



Ecosystem information (e.g. which fish eats which fish)



Commercial codes



General information, **occurrence** data, including information from other sources



General information, figures

Yannis Tzitzikas et al., ESWC 2014,  
Heraklion, Crete

# Marine Information: in several sources

Using and accessed through  
**different technologies**



Web services (SOAP/WSDL)



RDF + OWL files



SPARQL Endpoint



Relational Database



SPARQL Endpoint







# *How to integrate*

Yannis Tzitzikas et al., ESWC 2014,  
Heraklion, Crete

# Main approaches for Integration

In general there are two main approaches for integration

## **Warehouse** approach (materialized integration)

- Design Phase: The underlying sources (and their parts) have to be selected
- Creation Phase: Process for getting and creating the warehouse
- Maintenance Phase: Ability to create the warehouse from scratch, and/or ability to update parts of it
- **Mappings** are exploited to **extract** information from data sources, to **transform** it to the target model and then to **store** it at the central repository

## **Mediator** approach (virtual integration)

- The mediator receives a query formulated in terms of the unified model/schema. The **mappings** are used to enable **query translation**. The derived sub-queries are sent to the wrappers of the individual sources, which transform them into queries over the underlying sources. The results of these sub-queries are sent back to the mediator where they are assembled to form the final answer

# Main approaches for integration (cont.)

## Warehouse

- **Benefit: Flexibility in transformation logic** (including ability to curate and fix problems)
- **Benefit:** Decoupling of the release management of the integrated resource from the management cycles of the underlying sources
- **Benefit:** Decoupling of access load from the underlying sources.
- **Benefit: Faster responses** (in query answering but also in other tasks, e.g. if one wants to use it for applying an entity matching technique).
- **Shortcomings** You have to pay the cost for hosting the warehouse. You have to refresh periodically the warehouse

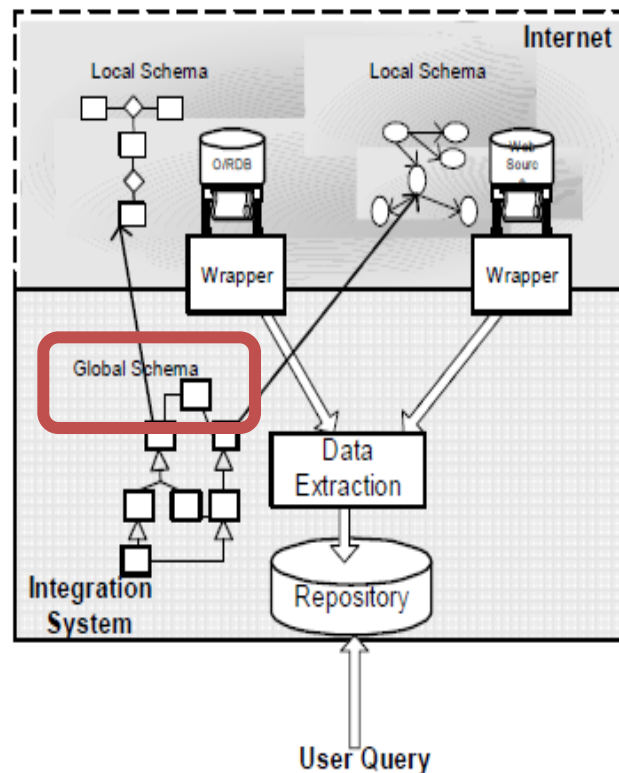
## Mediator

- **Benefit:** One advantage (but in some cases disadvantage) of **virtual integration** is the real-time reflection of **source updates in integrated access**
- Comment: The higher complexity of the system (and the quality of service demands on the sources) is only **justified if immediate access to updates is indeed required.**

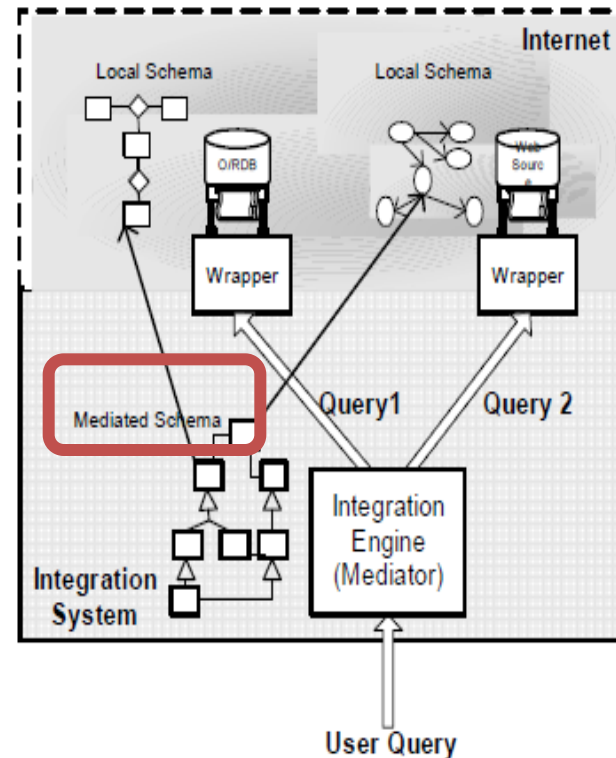
# Main approaches for integration (cont.)

In both cases we need a **unified model/schema**

## Materialized views



## Virtual views



# The Top Level Ontology: **MarineTLO**

**MarineTLO** aims at being a **global core model** that

- provides a **common, agreed-upon and understanding** of the **concepts** and **relationships** holding in the **marine domain** to enable knowledge sharing, information exchanging and integration between heterogeneous sources
- covers with **suitable abstractions** the marine domain to enable the most **fundamental queries**, can be extended to **any level of detail on demand**, and
- allows data originating from distinct sources to be adequately **mapped** and **integrated**
- **MarineTLO is not** supposed to be the **single** ontology covering the entirety of what exists

Benefits:

- **reduced effort** for **improving and evolving** : the focus is given on one model, rather than many (the results are beneficial for the entire community)
- **reduced effort** for constructing **mappings**: this approach avoids the inevitable combinatorial explosion and complexities that results from pair-wise mappings between individual metadata formats and/or ontologies

# MarineTLO: Query capabilities

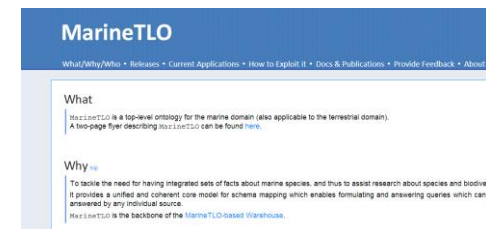
It allows formulating complex queries, e.g.:

1. Given the *scientific name* of a species, find its *predators* with the related *taxon-rank classification* and with the different *codes* that the organizations use to refer to them.

2. Given the *scientific name* of a species, find the *ecosystems*, *waterareas* and *countries* that this species is *native to*, and the *common names* that are *used* for this species *in each of the countries*

The MarineTLO currently contains around 90 classes and 40 properties.

More in [www.ics.forth.gr/is1/MarineTLO](http://www.ics.forth.gr/is1/MarineTLO)



The screenshot shows the MarineTLO website interface. At the top, there is a navigation menu with links: 'What/Why/Who', 'Releases', 'Current Applications', 'How to Exploit it', 'Docs & Publications', 'Provide Feedback', and 'About'. Below the menu, the 'What' section is visible, containing the text: 'MarineTLO is a top-level ontology for the marine domain (also applicable to the terrestrial domain). A two-page flyer describing MarineTLO can be found [here](#).' The 'Why' section follows, containing the text: 'To tackle the need for having integrated sets of facts about marine species, and thus to assist research about species and biodiversity it provides a unified and coherent core model for schema mapping which enables formulating and answering queries which cannot be answered by any individual source. MarineTLO is the backbone of the [MarineTLO-based Warehouse](#).'

# The MarineTLO-based semantic warehouse



# Requirements



# Requirements

## Functional Requirements

- F1: Multiplicity Of Sources
- F2: Mappings, Transformations and Equivalences
- F3: Reconstructibility

## Non Functional Requirements

- N1: Scope control
- N2: Connectivity assessment
- N3: Provenance
- N4: Consistency and Conflicts

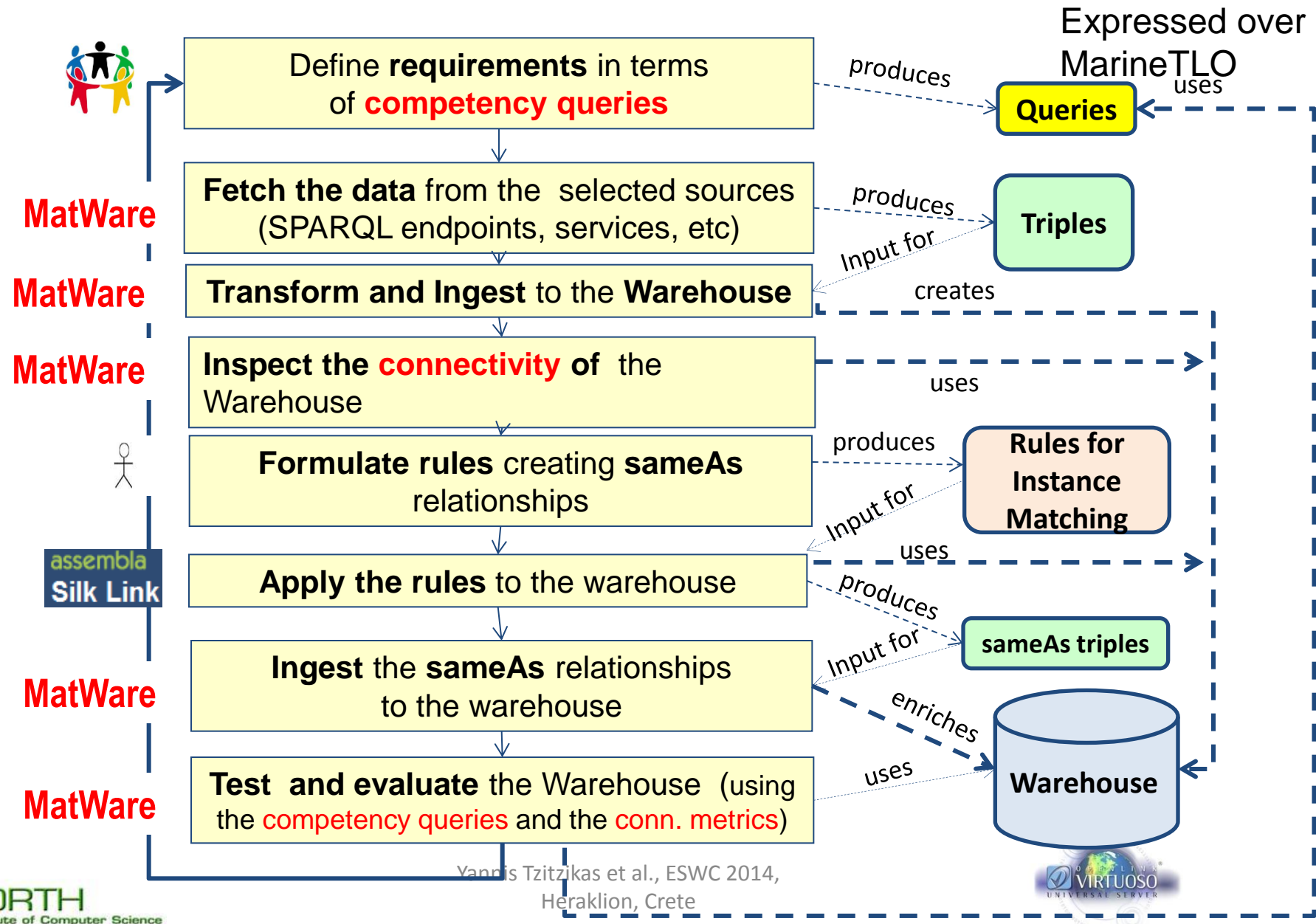
,

- **ODCleanStore** [T. Knap et al. 2012]
  - Downloading of RDF graphs
  - Deduplication
  - Conflict Resolution
    - Two sources (subjects) have different object values for a certain predicate
    - Rules for selecting one or more values (ANY, MAX, ALL, etc.)
  - Quality metrics for a) scoring a source based on conflicts, and b) assessing the overall outcome [T. Knap and J. Michelfeit, 2012]
- **Sieve** [P. N. Mendes et al. 2012]
  - Part of the Linked Data Integration Framework (LDIF)
  - Metrics for *schema completeness* and *conciseness*

# The warehouse construction tool

## **MatWare**

# The warehouse **construction** and **evolution** process (as supported by **MatWare**)

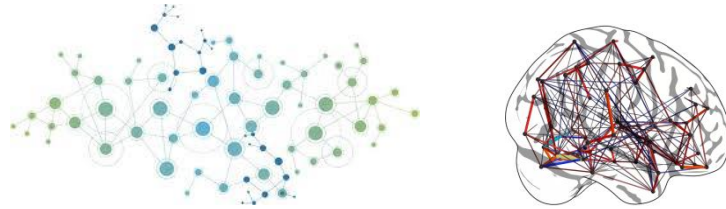


# N1: Scope Control

- We use the notion of **competency queries**.
  - A **competency query** is a query that is useful for the community at hand, e.g. for a human member , or for building applications for that domain
- Indicative competency queries for the warehouse of iMarine:

#Query	For a scientific name of a species (e.g. <i>Thunnus Albacares</i> or <i>Poromitra Crassiceps</i> ), find/give me:
Q <sub>1</sub>	the biological environments (e.g. <b>ecosystems</b> ) in which the species has been introduced and more general descriptive information of it (such as the <b>country</b> )
Q <sub>2</sub>	its <b>common names</b> and their complementary info (e.g. <b>languages</b> and <b>countries</b> where they are used)
Q <sub>3</sub>	the <b>water areas</b> and their <b>FAO codes</b> in which the species is native
Q <sub>4</sub>	the <b>countries</b> in which the species lives
Q <sub>5</sub>	the <b>water areas</b> and the <b>FAO portioning code</b> associated with a country
Q <sub>6</sub>	the presentation w.r.t <b>Country, Ecosystem, Water Area</b> and <b>Exclusive Economical Zone</b> (of the water area)
Q <sub>7</sub>	the projection w.r.t. <b>Ecosystem</b> and <b>Competitor</b> , providing for each competitor the <b>identification information</b> (e.g. several codes provided by different organizations)
Q <sub>8</sub>	a map w.r.t. <b>Country</b> and <b>Predator</b> , providing for each predator both the <b>identification information</b> and the <b>biological classification</b>
Q <sub>9</sub>	<b>who</b> discovered it, in which <b>year</b> , the <b>biological classification</b> , the <b>identification information</b> , the <b>common names</b> - providing for each common name the <b>language</b> and the <b>countries</b> where it is used in.

# N2: Connectivity Assessment



- **Connectivity** has two main aspects: **Schema** and **Instance**
- Regarding **Schema Connectivity** we use a **top level ontology (MarineTLO)** and **schema mappings** in order to associate the fetched data with the schema of the top level ontology.
  - Based on these MatWare transforms and then ingests the fetched data
- As regards **Instance Connectivity** one has to inspect and test the connectivity of the “draft” warehouse through the competency queries, and a number of connectivity metrics that we have defined and then formulate rules for **instance matching**

# N2: Connectivity Metrics

- Motivation: Why it is useful to measure Connectivity
  - For assessing **how much** the aggregated content is **connected**
  - For getting an **overview** of the warehouse
  - For **quantifying the value** of the warehouse (**query capabilities**)
    - Poor connectivity affects negatively the query capabilities of the warehouse.
  - For making **easier its monitoring** after reconstruction
  - For **measuring the contribution of each source** to the warehouse, and hence deciding which sources to keep or exclude (there are already hundreds of SPARQL endpoints). Identification of redundant or unconnected sources

MatWare supports the connectivity metrics introduced in :

- Y. Tzitzikas, et al, ***Quantifying the Connectivity of a Semantic Warehouse***, 4th International Workshop on Linked Web Data Management (LWDM'14@ EDBT'14)
- M. Mountantonakis et al, ***Extending VOID for Expressing the Connectivity Metrics of a Semantic Warehouse***, 1st International Workshop on Dataset Profiling & Federated Search for Linked Data (PROFILES'14), ESWC'14,

# N2: Connectivity Metrics Definition

Metric Name	Metric Definition
Common URIs between two Sources $S_i$ and $S_j$	$ U_i \cap U_j $
Percentage of Common URIs between two Sources $S_i$ and $S_j$	$curi_{i,j} = \frac{ U_i \cap U_j }{\min( U_i ,  U_j )}$
Common Literals between two Sources $S_i$ and $S_j$	$ Lit_i \cap Lit_j $
Percentage of Common Literals between two Sources $S_i$ and $S_j$	$clit_{i,j} = \frac{ Lit_i \cap Lit_j }{\min( Lit_i ,  Lit_j )}$
Increase in the average degree	$\frac{deg_W(E) - deg_S(E)}{deg_S(E)}$
Unique Triples of a Source $S_i$	$triplesUnique(S_i) = triples(S_i) \setminus (\cup_{1 \leq j \leq k, j \neq i} triples(S_j))$
Percentage of Unique Triples of a Source $S_i$	$\frac{ triplesUnique(S_i) }{ triples(S_i) }$
Complementarity factor for an entity $e$	$cf(e) =  \{ i \mid triples_W(e) \cap triplesUnique(S_i) \neq \emptyset \} $



## N2: Connectivity Metrics:

### Increase in the average Degree

$S_i$	$\text{avg } \text{deg}_{S_i}(U_i)$	$\text{avg } \text{deg}_W(U_i)$	increase
FLOD	7.18	9.18	27.84%
WoRMS	3.3	7.33	122.36%
Ecoscope	22.84	31.18	36.56%
DBpedia	41.41	42.11	1.7%
FishBase	18.86	29.81	58.08%
<b>AVERAGE</b>	18.72	23.92	27.78%

≡ Suffix canonicalization

The average degree is increased from 18.72 to 23.92.

Average degrees in sources and in the warehouse

$S_i$	$\text{avg } \text{deg}_{S_i}(U_i)$	$\text{avg } \text{deg}_W(U_i)$	increase
FLOD	7.18	54.31	656.51%
WoRMS	3.3	9.93	201.36%
Ecoscope	22.84	165.24	623.6%
DBpedia	41.41	84.2	103.36%
FishBase	18.86	50.6	168.32%
<b>AVERAGE</b>	18.72	72.86	289.21%

≡ Entity Matching

The average degree, of all sources is significantly bigger than before.

Average degrees in sources and in the warehouse

# N2: Connectivity Metrics:

As computed by MatWare

## Metrics Results

Produced by MatWare on: 1/12/2013

SPARQL EndPoint: <http://virtuoso.i-marine.d4science.org:8890/sparql>

Sources Used: i) FLOD ii) WoRMS iii) Ecoscope iv) DBpedia v) Fishbase vi) Clone Source vii) Airports

### Common Uris

Source	FLOD	WoRMS	Ecoscope	DBpedia	Fishbase	Clone Source	Airports
FLOD	173929	239	523	631	887	250	13
WoRMS		80485	200	1714	3596	364	0
Ecoscope			5824	192	225	4030	4
DBpedia				70246	9578	4589	14
Fishbase					34974	481	60
Clone Source						8457	4
Airports							4606

### Common Uris Percentage

Source	FLOD	WoRMS	Ecoscope	DBpedia	Fishbase	Clone Source	Airports
FLOD	1	0.3%	8.98%	0.9%	2.54%	2.96%	0.28%
WoRMS		1	3.43%	2.44%	10.28%	4.3%	0%
Ecoscope			1	3.3%	3.86%	69.2%	0.09%
DBpedia				1	27.39%	54.26%	0.3%
Fishbase					1	5.69%	1.3%
Clone Source						1	0.09%
Airports							1

### Common Literals

Source	FLOD	WoRMS	Ecoscope	DBpedia	Fishbase	Clone Source	Airports
FLOD	111164	3624	1745	5668	9504	373	1533
WoRMS		51076	382	2429	4773	289	86
Ecoscope			14102	389	422	6871	131
DBpedia				123887	14038	7144	117
Fishbase					138275	604	152
Clone Source						13964	49
Airports							12302

### Common Literals Percentage

Source	FLOD	WoRMS	Ecoscope	DBpedia	Fishbase	Clone Source	Airports
FLOD	1	7.1%	12.37%	5.1%	8.55%	2.67%	12.46%
WoRMS		1	2.71%	4.76%	9.34%	2.07%	0.7%
Ecoscope			1	2.76%	2.99%	49.21%	1.06%
DBpedia				1	11.33%	51.16%	0.95%
Fishbase					1	4.33%	1.24%
Clone Source						1	0.4%
Airports							1

### Triples

Source	Triples	Unique Triples	Percentage
FLOD	665456	664703	99.89%
WoRMS	461230	460741	99.89%
Ecoscope	54027	17951	33.23%
DBpedia	450429	429426	95.34%
Fishbase	1425283	1424713	99.96%
Clone Source	56166	0	0%
Airports	31628	31628	100%

\* Probably redundant source

### Complementarity Factor

Entities	Complementarity Factor
Astrapogon	2 7
Species	5 7
Greece	4 7
Thunnus	5 7
Shark	5 7

### Degrees

Source	Source Degree	Warehouse Degree	Increase
FLOD	7.18	54.3	656.4%
WoRMS	3.3	9.93	200.09%
Ecoscope	22.84	165.24	623.46%
DBpedia	41.41	84.2	104.8%
Fishbase	18.86	50.6	168.29%
Clone Source	44.43	84.2	89.5%
Airports	70.99	72.56	2.2%
Average	41.8	74.43	78.07%

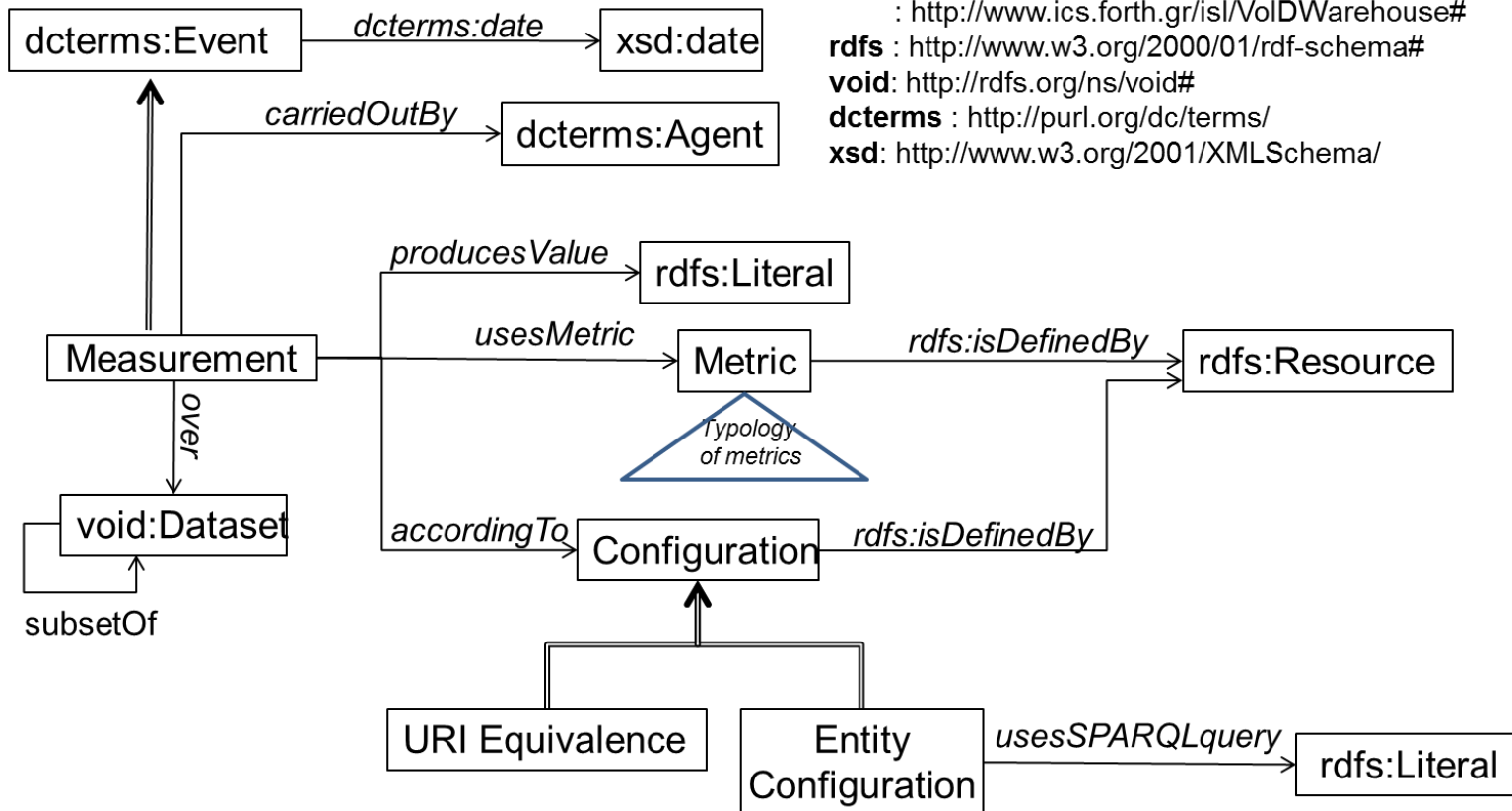
\* Probably out of domain of interest

# N2: : Connectivity Metrics: *Exchanging*

We have extended VoID for representing and exchanging such metrics  
(VoIDWarehouse)

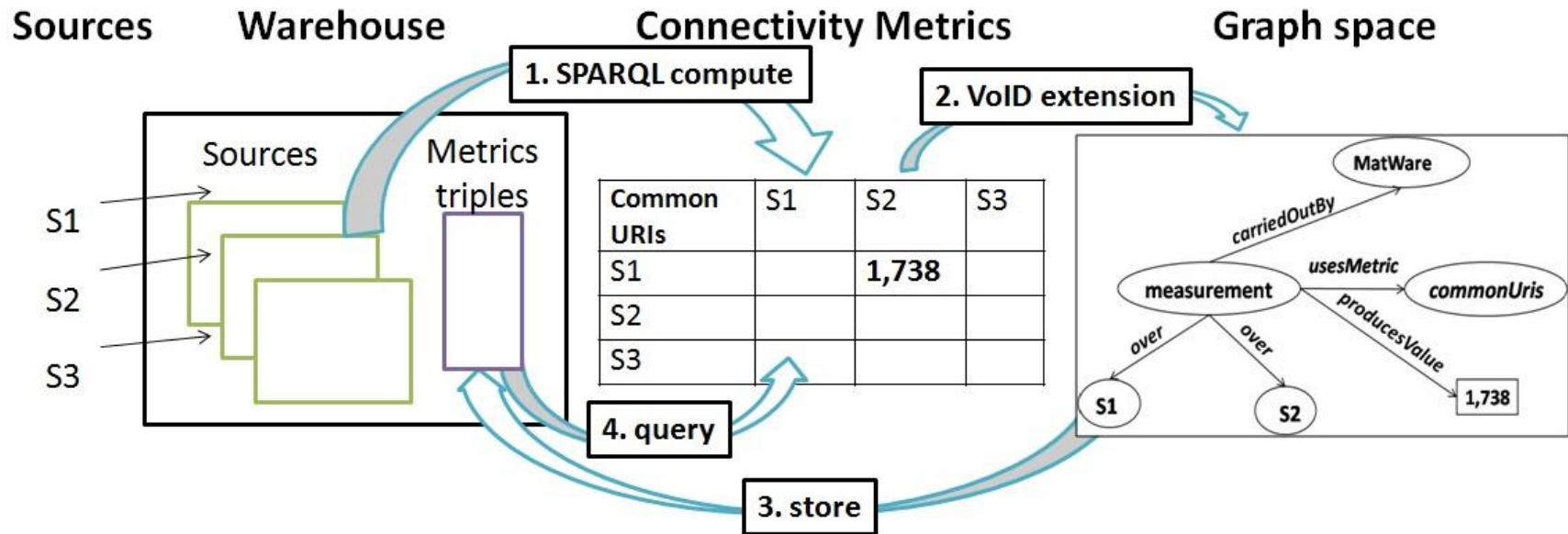
## Namespaces

`ics` : <http://www.ics.forth.gr/is1/VoIDWarehouse#>  
`rdfs` : <http://www.w3.org/2000/01/rdf-schema#>  
`void` : <http://rdfs.org/ns/void#>  
`dcterms` : <http://purl.org/dc/terms/>  
`xsd` : <http://www.w3.org/2001/XMLSchema/>



Published in: <http://www.ics.forth.gr/is1/VoIDWarehouse>

# N2: : Connectivity Metrics: *Exchanging* (cont).



1. **Compute** of the Connectivity Metrics-Production of Matrixes
2. **Describe** the Connectivity Metrics with the proposed VoID extension
3. **Store** these triples in a separate graph space
4. **Retrieve/Query** these values from the warehouse using SPARQL queries

# N3: Provenance

We have realized that the following 4 levels of provenance support are usually required:

- [a] **Conceptual level**
- [b] **URIs and Values level**
- [c] **Triple Level**
- [d] **Query level**

Level [a] can be supported by the conceptual model level. In our application context we use the MarineTLO and the transformation rules do the required transformations.

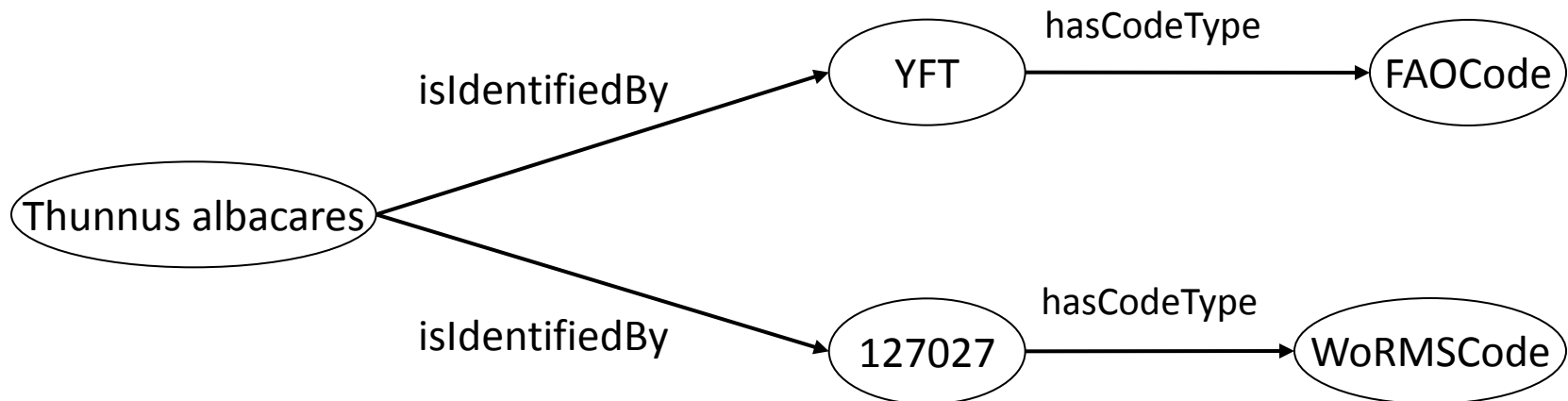
Matware offers support also for levels [b]-[d]

# N3: Provenance

## a) Conceptual modeling level

Example: *Assignment of identifiers to species*

- MarineTLO models the provenance of species names, codes etc, and the Transformation rules of MatWare transform the ingested data according to this model.



# N3: Provenance

## b) URIs and Literals :

- i. Adopting the namespace mechanism for **URIs**:
  - The prefix of the URI provides information about the origin of the data.
  - e.g. [www.fishbase.org/entity/ecosystem#mediteranean\\_sea](http://www.fishbase.org/entity/ecosystem#mediteranean_sea)
- ii. Ability to attach @Source to every **literal** coming from a Source:
  - e.g. select scientific name and authorship of Yellow Fin Tunna

scientificName	year	authorship
"Thunnus albacares"@dbpedia	"1788"@dbpedia	"Bonnaterre"@dbpedia
"Thunnus albacares"@worms	"1788"@worms	"Bonnaterre"@worms

- This policy allows formulating source-centric queries in a relative simple way:

```
SELECT ?speciesname
WHERE {
  ?species tlo:has_scientific_name ?scientificname
  FILTER(langMatches(lang(?scientificname), "worms"))
}
```

## c) Triple Level Provenance

- Store the fetched triples in a **separate graphspace**:
  - *FISHBASE*: <http://www.ics.forth.gr/isl/Fishbase>
  - *DBpedia*: <http://www.ics.forth.gr/isl/DBpedia>
  - *FLOD*: <http://www.ics.forth.gr/isl/FLOD>
  - *Ecoscope*: <http://www.ics.forth.gr/isl/Ecoscope>
  - *WoRMS*: <http://www.ics.forth.gr/isl/WoRMS>
- By asking for the graph that each triple is coming from we retrieve the provenance of the data.



## d) Query Level Provenance

- Matware offers a **query rewriting** functionality that exploits the contents of the graphspaces for returning the sources that contributed to the query results (including those that contributed to the intermediate steps).
- Let  $q$  be a SPARQL query that has  $n$  parameters in the select clause and contains  $k$  triple patterns of the form  $(?s_i, ?p_i, ?o_i)$  :

```
SELECT {?o_1 ?o_2} WHERE {  
  ?s_1 ?p_1 ?o_1 .  
  ?s_2 ?p_2 ?o_2 .  
  ?s_k ?p_k ?o_k  
}
```

- The rewriting produces a query  $q'$  that has  $n+k$  parameters in the select clause and each triple pattern  $(?s_i ?p_i ?o_i)$  has been replaced by: `graph ?gi {?si ?pi ?oi}`. Eventually the rewritten query  $q'$  is:

```
SELECT {?o_1 ?o_2 ?g_1 ?g_2 ?g_k} WHERE {  
  graph ?g_1 {?s_1 ?p_1 ?o_1 }.  
  graph ?g_2 { ?s_2 ?p_2 ?o_2 } .  
  graph ?g_3 {?s_k ?p_k ?o_k}  
}
```

# N3: Provenance

## Query Level Provenance Example:

**QUERY:** For a scientific name of a species (e.g. *Thunnus Albacares*) find the FAO codes of the waterareas in which the species is native.

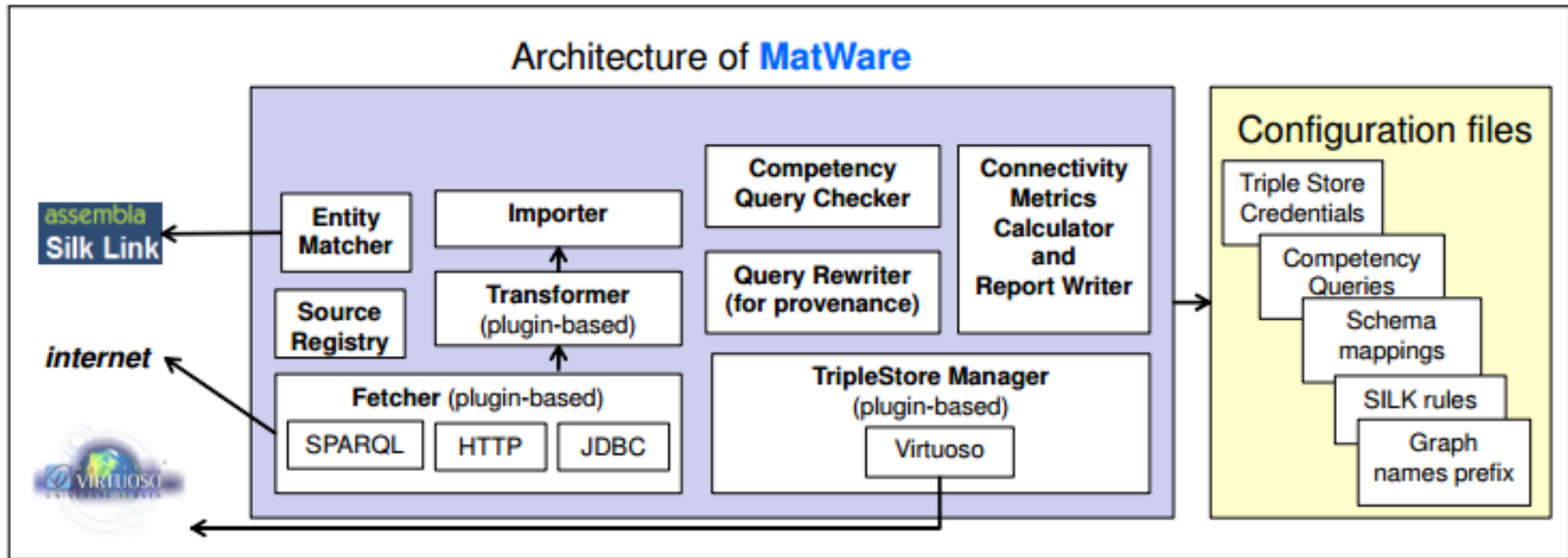
### SPARQL:

```
select ?faocode ?source1 ?source2
where {
  graph ?source1 {
    ecoscope:thunnus_albacares MarineTLO:isNativeAt ?waterarea
  }.
  graph ?source2 {
    ?waterarea MarineTLO:LXrelatedIdentifierAssignment ?faocode
  }
}
```

### RESULT:

faocode	source1	source2
47	<a href="http://www.ics.forth.gr/isl/Fishbase">http://www.ics.forth.gr/isl/Fishbase</a>	<a href="http://www.ics.forth.gr/isl/FLOD">http://www.ics.forth.gr/isl/FLOD</a>
34	<a href="http://www.ics.forth.gr/isl/Fishbase">http://www.ics.forth.gr/isl/Fishbase</a>	<a href="http://www.ics.forth.gr/isl/FLOD">http://www.ics.forth.gr/isl/FLOD</a>
31	<a href="http://www.ics.forth.gr/isl/Fishbase">http://www.ics.forth.gr/isl/Fishbase</a>	<a href="http://www.ics.forth.gr/isl/FLOD">http://www.ics.forth.gr/isl/FLOD</a>
41	<a href="http://www.ics.forth.gr/isl/Fishbase">http://www.ics.forth.gr/isl/Fishbase</a>	<a href="http://www.ics.forth.gr/isl/FLOD">http://www.ics.forth.gr/isl/FLOD</a>
71	<a href="http://www.ics.forth.gr/isl/Fishbase">http://www.ics.forth.gr/isl/Fishbase</a>	<a href="http://www.ics.forth.gr/isl/FLOD">http://www.ics.forth.gr/isl/FLOD</a>
77	<a href="http://www.ics.forth.gr/isl/Fishbase">http://www.ics.forth.gr/isl/Fishbase</a>	<a href="http://www.ics.forth.gr/isl/FLOD">http://www.ics.forth.gr/isl/FLOD</a>

# Architecture of Matware

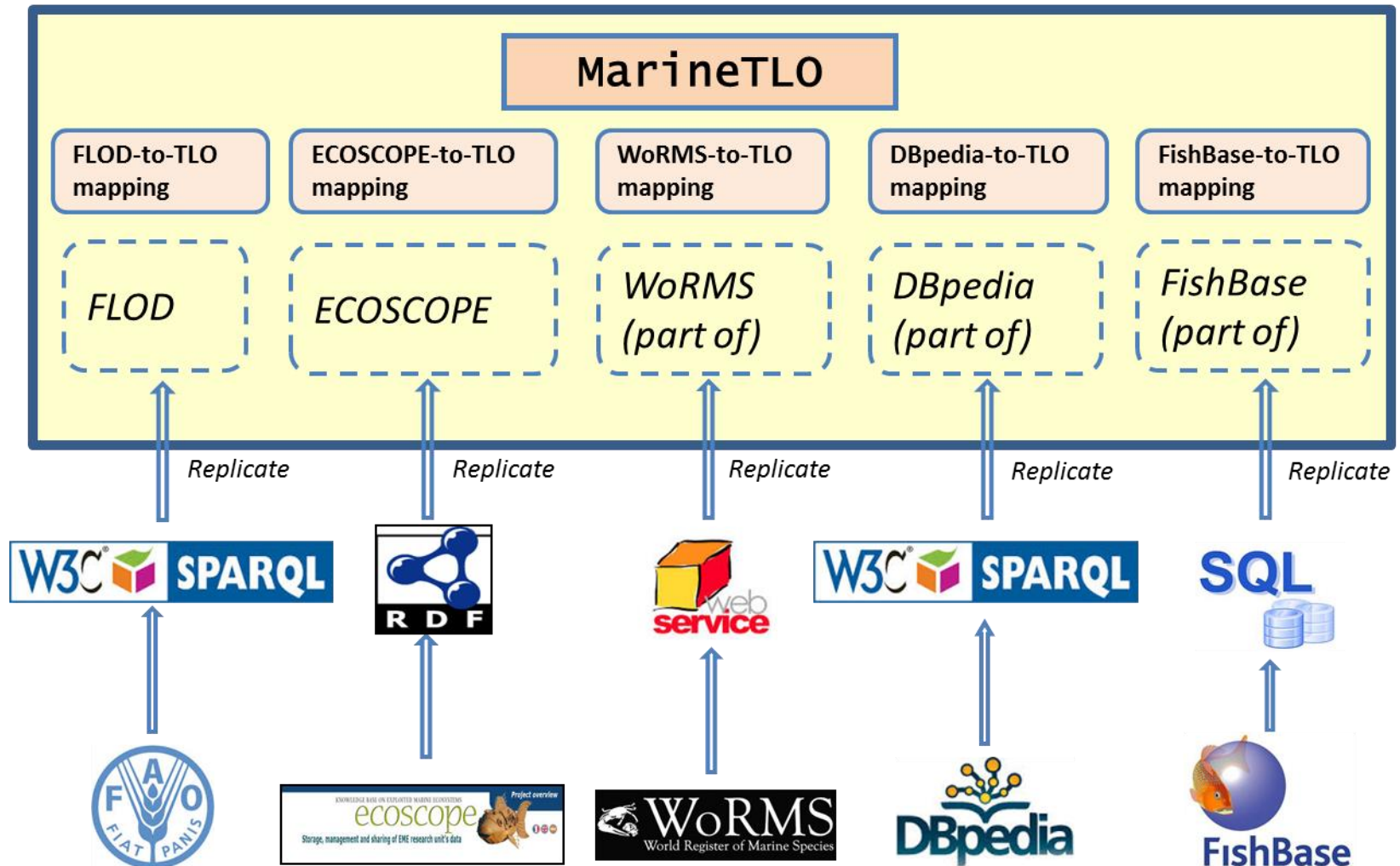


- **Actions in order to create a Warehouse from scratch** one should specify
  - the type of the repository
  - the names of the graphs that correspond to the different sources
  - URL, username and password in order to connect to the repository
- **Actions in order to add a new source**
  - (a) include the fetcher class for the specific source as plug in
  - (b) provide the mapping files (schema mappings)
  - (c) include the transformer class for the specific source as a plug in
  - (d) provide the SILK rules as xml files

# Using MatWare for constructing the MarineTLO-based warehouse

# The MarineTLO-based semantic warehouse

MatWare



Yannis Tzitzikas et al., ESWC 2014,  
Heraklion, Crete

# The resulted **MarineTLO**-based Warehouse(3/3)

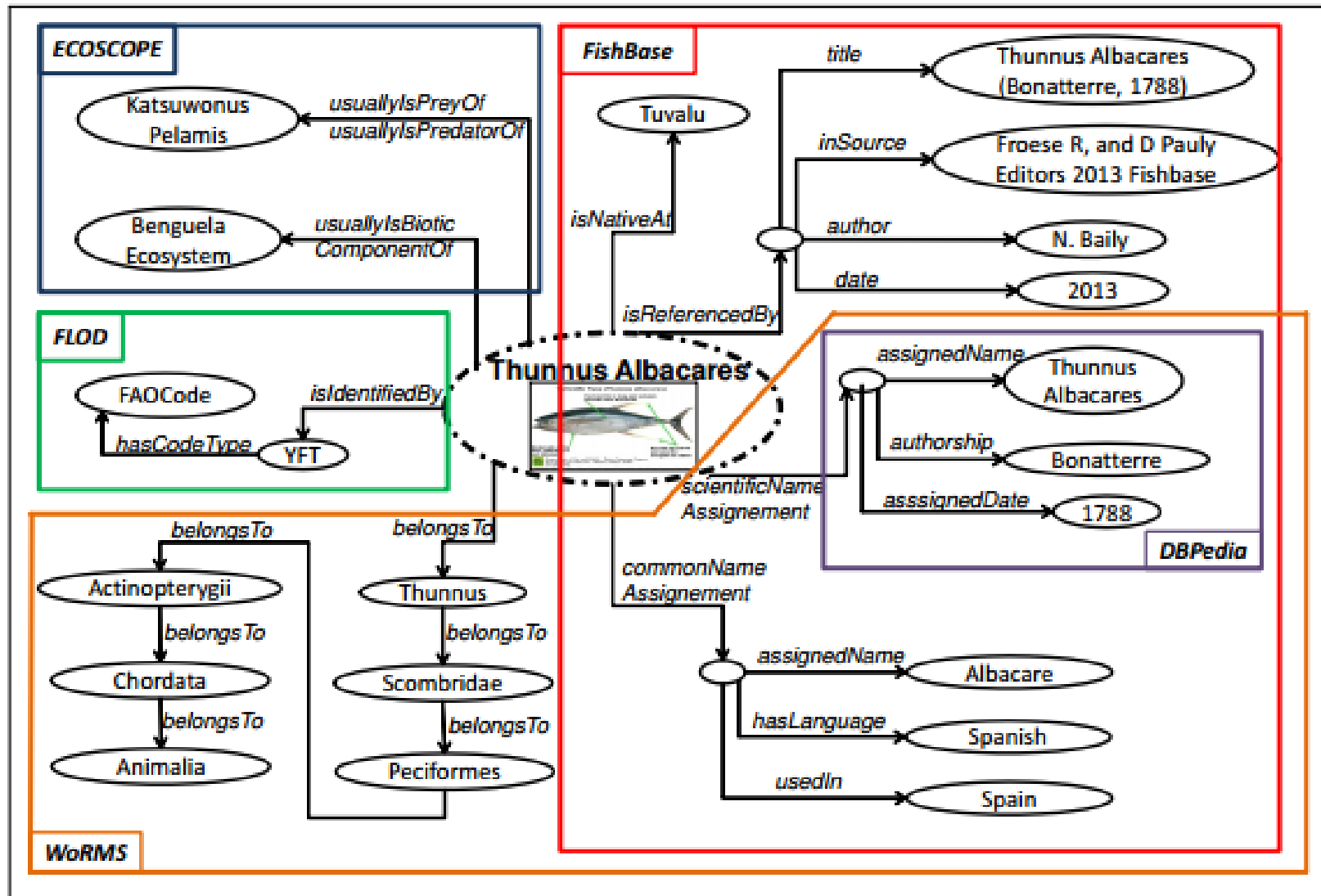
## The current warehouse contains:

- 37,000 marine species.
- 3,772,919 triples

Process	Time (Min)
Downloading the Sources	60
Importing the data	15
Applying the transformation rules	40
Producing sameAs links using SILK	30
Computing the metrics	50
<b>Total Process</b>	<b>195</b>

# The resulted **MarineTLO**-based Warehouse(1/3)

Integrated information about *Thunnus albacares* from different sources



# The resulted **MarineTLO**-based Warehouse(2/3)

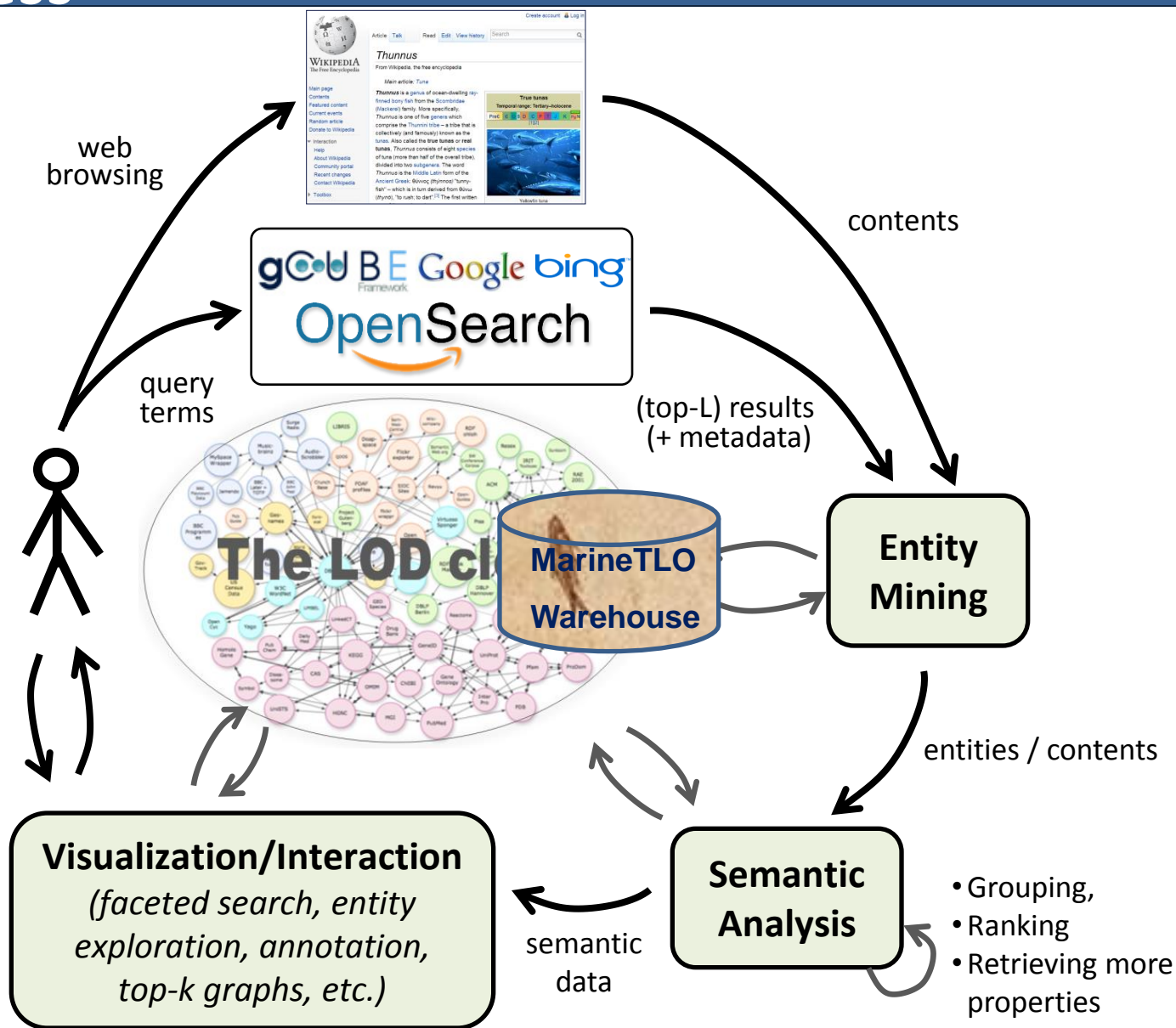
Concepts	Ecoscope	FLOD	WoRMS	DBpedia	Fishbase
Species	✓	✓	✓	✓	✓
Scientific Names	✓	✓	✓	✓	✓
Authorships			✓	✓	✓
Common Names	✓	✓	✓	✓	✓
Predators	✓				✓
Ecosystems	✓				✓
Countries					✓
Water Areas		✓			✓
Vessels	✓	✓			✓
Gears	✓	✓			✓
EEZ		✓			



# Applications that use this Warehouse

- 1/ Semantic Post-Processing of Search Results (e-infrastructure service)
- 2/ Fact Sheet Generator (web application)
- 3/ Ichthys Android app

# For Semantic Post-Processing of Search Results: The process



# For Semantic Post-Processing of Search Results: Example (X-Search)

**Semantically Enriched Results**

Query: tuna  
In Collections: FIGIS

**Mined Entities**

- FAOCountry(24)
  - Republic of ...(1)
  - Viet nam(1)
  - Venezuela(2)
  - Yugoslavia(2)
  - Senegal(1)
- Species(8)
  - eastern Paci...(1)
  - yellowtail a...(1)
  - Ara(1)
  - pantropical...(1)
  - Indo-Pacific...(1)
- WaterAreas(3)
  - Mediterranea...(1)
  - Atlantic(2)

**Object Metadata**

[Thunnus albacares \(Bonnaterre, 1788\) - Fact sheet](#)

Yellowfin **tuna**... (Venezuela), Ca bo Vang (Viet nam), **Tuna** zutoperka (Yugoslavia)... There are important yellowfin **tuna** fisheries throughout tropical and subtropical seas. The most... major surface fishing techniques for yellowfin **tuna** in the Pacific, even though this method is... **tuna** in the Pacific is nearly contin... [show all](#)

**Textual Clustering**

- Root(15)
  - fact sheet(1)
  - thunnus(8)
  - stenella(4)
  - linnaeus(1)
  - axis(1)
  - max fact s...(1)
  - purus lin...(1)
  - rda chille...(1)

**Semantic Entity Exploration**

- URI: <http://www.fao.org/figis/flod/entities/codedentity/3e6d22db-1f06-437d-ac4a-9d3c8b895bf5> (open)
- Value: yellowtail amberjack

---

- URI: [http://dbpedia.org/resource/Yellowtail\\_amberjack](http://dbpedia.org/resource/Yellowtail_amberjack) (open)
- Value: Yellowtail amberjack

**Semantic Entity Exploration**

Properties of: [Yellowtail\\_amberjack](#)

<b>Type</b>	<b>SameAs</b>		
<ul style="list-style-type: none"> <li>Animal (open)</li> <li>Thing (open)</li> <li>Species (open)</li> <li>FLODSpecies (open)</li> <li>Fish (open)</li> <li>CodedEntity (open)</li> <li>Eukarvotte (open)</li> <li>Animal (open)</li> <li>Fish (open)</li> </ul>	<ul style="list-style-type: none"> <li>Seriola lalandi (open)</li> </ul>		
<b>Subject</b>	<b>Class</b>		
<ul style="list-style-type: none"> <li>Category:Fish of the Red Sea (open)</li> <li>Category:Fish of the Indian Ocean (open)</li> <li>Category:Seriola (open)</li> </ul>	<ul style="list-style-type: none"> <li>Actinopterygii (open)</li> </ul>		
<b>BinomialAuthority</b>	<b>Family</b>		
<ul style="list-style-type: none"> <li>Georges Cuvier (open)</li> <li>Achille Valenciennes (open)</li> </ul>	<ul style="list-style-type: none"> <li>Carangidae (open)</li> </ul>		
<b>Genus</b>	<b>Kingdom</b>	<b>Order</b>	<b>Phylum</b>
<ul style="list-style-type: none"> <li>Seriola (open)</li> </ul>	<ul style="list-style-type: none"> <li>Animal (open)</li> </ul>	<ul style="list-style-type: none"> <li>Perciformes (open)</li> </ul>	<ul style="list-style-type: none"> <li>Chordate (open)</li> </ul>
<b>Depiction</b>	<b>Thumbnail</b>		
<ul style="list-style-type: none"> <li>Seriola lalandi.jpg (open)</li> </ul>	<ul style="list-style-type: none"> <li>200px-Seriola lalandi.jpg (open)</li> </ul>		

The Warehouse is used

The Warehouse is used

Result of Entity Mining

Result of Textual Clustering

The Warehouse is used

## Thunnus Albacares - Entity Exploration (close)

### Comment

• The yellowfin tuna (*Thunnus albacares*) is a species of tuna found in pelagic waters of tropical and subtropical oceans worldwide. Yellowfin is often marketed as ahi, from its Hawaiian name 'ahi although the name 'ahi in Hawaiian also refers to the closely related bigeye tuna. The species name, albacares (white meat) can lead to confusion. The tuna known as albacore in English, is a different species of tuna: *Thunnus alalunga*.

### Binomial Authority

• Pierre\_Joseph\_Bonnaterre (open)

### Class

• Actinopterygii (open)

### Family

• Scombridae (open)

### Genus

• Thunnus (open)

### Kingdom

• Animal (open)

### Order

• Perciformes (open)

### Phylum

• Chordate (open)

### FAO Distribution

• Species\_dist.22559 (open)  
• Species\_dist.22560 (open)

### Is Classified By Code

• Yft (open)

### ScientificName

• Thunnus albacares

### Has Related Indicator

• Kernel\_density (open)  
• I3\_SpeciesYearByGearMonth\_YFT\_1999 (open)  
• I3\_SpeciesYearByGearMonth\_YFT\_2001 (open)  
• I2\_SpeciesByGear\_YFT (open)  
• I3\_SpeciesYearByGearMonth\_YFT\_2000 (open)  
• I6\_SpeciesMap\_YFT\_2001\_2002 (open)  
• I1\_SpeciesByOcean\_YFT (open)

### Is Predator Of

• Sarda\_orientalis (open)  
• Sthenoteuthis\_oualaniensis (open)  
• Mastigoteuthis\_sp (open)  
• Scomberesox\_saurus (open)  
• Neoanchisquilla\_tuberculata (open)  
• Vinciguerra\_nimbaria (open)  
• Odontodactylus\_scyllarus (open)  
(show all)

### Depiction

• IndicatorThunnusAlbacares2 (open)  
• IndicatorThunnusAlbacares1 (open)  
• IndicatorThunnusAlbacares4 (open)  
• IndicatorThunnusAlbacares3 (open)  
• PictureToBeTagged13\_1 (open)

### Related Layer

• LayerDbStomacYFT (open)  
• LayerDbSardaraYFT (open)

### Has Taxon Id

• WoRMS:127027 (open)

### Belongs To

• Thunnus (open)

Example of an EntityCard of Xsearch (if the entity's type is Species)

From DBpedia

From FLOD

From Ecoscope

From WoRMS

# XSearch as a bookmarklet

The Warehouse is used

## Annotating entities over the original page

WIKIPEDIA  
Free Encyclopedia

Navigation

Main page

Contents

Featured content

Current events

Random article

Donate to Wikipedia

Interaction

Help

About Wikipedia

Community portal

Recent changes

Contact Wikipedia

Toolbox

What links here

Related changes

Upload file

Special pages

Permanent link

Page information

Cite this page

Print/export

Create a book

Download as PDF

Printable version

## Thunnus

From Wikipedia, the free encyclopedia

*Main article: Tuna*

**Thunnus** is a genus of ocean-dwelling ray-finned bony fish from the **Scombridae** (Mackerel) family. More specifically, **Thunnus** is one of five genera which comprise the **Thunnini** tribe – a tribe that is collectively (and famously) known as the **tunas**. Also called the **true tunas** or **real tunas**, **Thunnus** consists of eight species of tuna (more than half of the overall tribe), divided into two subgenera. The word **Thunnus** is the Middle Latin form of the Ancient Greek: **θύσος** (*thýsōs*) "tunny-fish" – which is in turn derived from **θύω** (*thynō*), "to rush; to dart".<sup>[3]</sup> The first written use of the word was by Homer.<sup>[citation needed]</sup>

Their coloring, metallic blue on top and shimmering silver-white on the bottom, helps camouflage them from above and below. They can grow to 15 feet long and weigh over 1,000 pounds, and can swim up to 50 miles per hour when pursuing prey. **Atlantic** bluefin tunas are warm-blooded, which is a rare trait among fish, and are comfortable in the cold waters. Bluefin fish are found in Newfoundland and **Iceland**, as well as the tropical waters of the Gulf of **Mexico** and the **Mediterranean** Sea, where they go each year to spawn.

Due to overfishing the genus range has been significantly reduced, being effectively removed from the **Black Sea**, for example.<sup>[4]</sup>

## Taxonomy [edit]

This genus has eight species in two subgenera:

- Subgenus **Thunnus** (**Thunnus**):
  - Albacore**, *T. alalunga* (Bonnaterre, 1788).
  - Southern bluefin tuna**, *T. maccoyii* (Castelnau, 1872).
  - Bigeye tuna**, *T. obesus* (Lowe, 1839).
  - Pacific** bluefin tuna
  - Atlantic** bluefin tuna
- Subgenus **Thunnus** (**Thunnus**):
  - Yellowfin** tuna
  - Blackfin** tuna
  - Longtail** tuna

## Semantic Entity Enrichment

- URI**: [http://www.ecoscope.org/ontologies/ecosystem/s#thunnus\\_obesus](http://www.ecoscope.org/ontologies/ecosystem/s#thunnus_obesus) ([open](#))
- Label**: Bigeye tuna

Yannis Tzitzikas et al. LWDM 2014, Athens

## True tunas

Temporal range: Tertiary–holocene

PreЄ Є O S D C P T J K PgN  
[1][2]



**Yellowfin** tuna

## Scientific classification ✎

Kingdom: Animalia  
Phylum: Chordata  
Class: Actinopterygii  
Order: **Perciformes**  
Family: **Scombridae**  
Tribe: Thunnini  
Genus: **Thunnus**  
South, 1845

## Subgenus

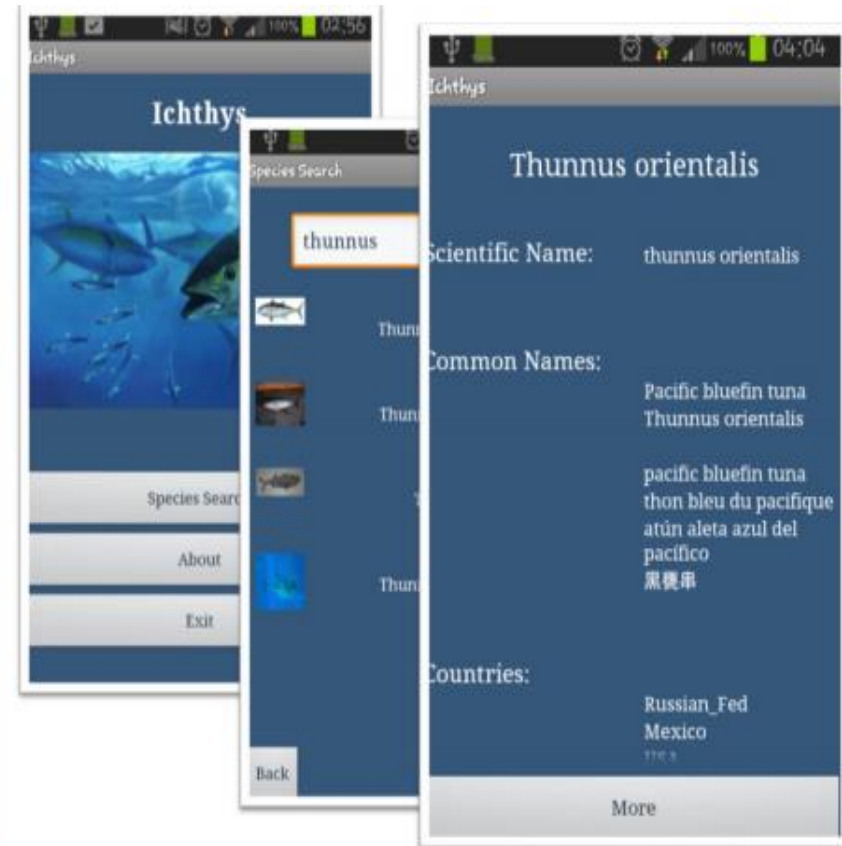
- T.* (**Thunnus**) (bluefin group)
- T.* (**Neothunnus**) (**yellowfin** group)

Entity exploration

# Fact Sheet Generator & Android Application



Fact Sheet Generator



Ichthys

# Concluding Remarks

- We have described the main requirements and challenges in designing, building, maintaining and evolving a real and operational semantic warehouse for marine resources
- We have presented the process and the tool **MatWare** that we have developed for supporting this process with emphasis on:
  - Scope control
  - Connectivity assessment
  - Provenance
  - Reconstructability
  - Extensibility

# Future Work

- Next steps
  - Further work on metrics for monitoring and understanding the **dynamics of the warehouse**
  - Investigate various **optimizations** for being scalable to very large amounts of data, relating inference materialization and provenance
    - E.g. using a single graph space that materializes all inferred triples for offering efficient query answering, while keeping also the separate graph spaces for provenance reasons



*Thank you for your attention*

Visit and send us feedback:

[www.ics.forth.gr/is1/MarineTLO](http://www.ics.forth.gr/is1/MarineTLO)

[www.ics.forth.gr/is1/Matware](http://www.ics.forth.gr/is1/Matware)

