D7.6.3 Third Prototype of the Fisheries Stock Depletion Assessment System (FSDAS)

Deliverable Co-ordinator: Claudio Baldassarre

Deliverable Co-ordinating Institution: FAO

Other Authors: Claudio Baldassarre (FAO), Germán Herrero Cárcel (ATOS), Michael Erdmann (ONTO), Aldo Gangemi (CNR)

This deliverable describes the finalization of the works implemented in the scope of T7.6 "Prototype of the Fisheries Stock Depletion Assessment System (FSDAS)". In particular we present: (a) the realization of web services to support FSDAS developers in populating the system knowledge base; (b) the design and implementation of software procedures that triple (i.e. produces RDF triples) the content of the Fisheries legacy sources. (c) the updated version of the FSDAS web client including all the fixes suggested by the users' feedback, and reported in [D7.7.2].

Document Identifier:	NEON/2010/D7.6.3/v1.0	Date due:	January 31st, 2010
Class Deliverable:	NEON EU-IST-2005-027595	Submission date:	January 31st, 2010
Project start date:	March 1, 2006	Version:	v1.0
Project duration:	4 years	State:	Final
		Distribution:	Public

NeOn Consortium

This document is a part of the NeOn research project funded by the IST Programme of the Commission of the European Communities by the grant number IST-2005-027595. The following partners are involved in the project:

Open University (OU) – Coordinator Knowledge Media Institute – KMi Berrill Building, Walton Hall Milton Keynes, MK7 6AA United Kingdom Contact person: Enrico Motta E-mail address: e.motta@open.ac.uk	Universität Karlsruhe – TH (UKARL) Institut für Angewandte Informatik und Formale Beschreibungsverfahren – AIFB Englerstrasse 11 D-76128 Karlsruhe, Germany Contact person: Andreas Harth E-mail address: aha@aifb.uni-karlsruhe.de
Universidad Politécnica de Madrid (UPM) Campus de Montegancedo 28660 Boadilla del Monte Spain Contact person: Asunción Gómez Pérez E-mail address: asun@fi.upm.es	Software AG (SAG) Uhlandstrasse 12 64297 Darmstadt Germany Contact person: Walter Waterfeld E-mail address: walter.waterfeld@softwareag.com
Intelligent Software Components S.A. (ISOCO) Calle de Pedro de Valdivia 10 28006 Madrid Spain Contact person: Jesús Contreras E-mail address: jcontreras@isoco.com	Institut 'Jožef Stefan' (JSI) Jamova 39 SI-1000 Ljubljana Slovenia Contact person: Marko Grobelnik E-mail address: marko.grobelnik@ijs.si
Institut National de Recherche en Informatique et en Automatique (INRIA) ZIRST – 655 avenue de l'Europe Montbonnot Saint Martin 38334 Saint-Ismier France Contact person: Jérôme Euzenat E-mail address: Jerome.euzenat@inrialpes.fr	University of Sheffield (USFD) Dept. of Computer Science Regent Court 211 Portobello street S14DP Sheffield United Kingdom Contact person: Hamish Cunningham E-mail address: hamish@dcs.shef.ac.uk
Universität Koblenz-Landau (UKO-LD) Universitätsstrasse 1 56070 Koblenz Germany Contact person: Steffen Staab E-mail address: staab@uni-koblenz.de	Consiglio Nazionale delle Ricerche (CNR) Institute of cognitive sciences and technologies Via S. Martino della Battaglia, 44 - 00185 Roma-Lazio, Italy Contact person: Aldo Gangemi E-mail address: aldo.gangemi@istc.cnr.it
Ontoprise GmbH. (ONTO) Amalienbadstr. 36 (Raumfabrik 29) 76227 Karlsruhe Germany Contact person: Jürgen Angele E-mail address: angele@ontoprise.de	Food and Agriculture Organization of the United Nations (FAO) Viale delle Terme di Caracalla 1 00100 Rome Italy Contact person: Caterina Caracciolo E-mail address: Caterina.Caracciolo@fao.org
Atos Origin S.A. (ATOS) Calle de Albarracín, 25 28037 Madrid, Spain Contact person: Tomás Pariente Lobo E-mail address: tomas.parientelobo@atosorigin.com	Laboratorios KIN, S.A. (KIN) C/Ciudad de Granada, 123 08018 Barcelona, Spain Contact person: Antonio López E-mail address: alopez@kin.es

Work package participants

The following partners have taken an active part in the work leading to the elaboration of this document, even if they might not have directly contributed writing parts of this document:

FAO, ATOS, ONTO, CNR

Change Log

Version	Date	Amended by	Changes
0.5	08/01/2010	Claudio Baldassarre	Chapter from 1 to 3. Section 4.1 and 4.1. Introduction to chapter 6
0.6	12/01/2010	German Herrero	Chapter 6 and Annex 10
0.7	13/01/2010	Aldo Gangemi	Section 4.3 4.4
0.8	15/01/2010	Michael Erdmann	Chapter 5
0.9	18/01/2010	Claudio Baldassarre	Conclusions
0.9a	19/01/2010	Michael Erdmann	Proof reading chapters 5, 6, and 7
0.9b	27/01/2010	Claudio Baldassarre	QA comments addressed
0.9c	02/02/2010	Caterina Caracciolo	Added ref. to D7.2.2. Added references to Sec. 2.3.1.
1.0	03/02/2010	Aneta Tumilowicz	Final QA

Executive Summary

The present deliverable contains work carried out in the context of Workpackage 7: "Ontology-driven fish stock depletion assessment system". The objectives of Workpackage 7 are (i) to create and maintain distributed ontologies in the fishery knowledge community, by deploying NeOn technologies (in a requirement/feedback loop); and (ii) to exploit ontologies within web applications and implement an assessment system.

This deliverable describes the finalization of the work implemented in the scope of T7.6 "Prototype of the Fisheries Stock Depletion Assessment System (FSDAS)", called FSDAS v3. In the past three years of the NeOn project T7.5 first, and T7.6 later, provided respectively the design and the implementation base lines to develop similar frameworks. These two tasks have been preceded by the formalization of the specifications collected from experts of the Fisheries department in FAO (T.7.1). Moreover assessment activity has been undertaken every time a new version of the FSDAS prototype was released (T.7.7). These works have been reported in their final form in previous related deliverables [D7.1.2], [D7.5.1], [D7.7.1] and [D7.7.2].

The finalisation of the work on FSDAS is entirely based on the user study undertaken in [D7.7.2]. The feedback from the users has driven all of the modifications, updates, and bug fixes to FSDAS v3. In particular we present:

- The realization of web services to support FSDAS developers in populating the system knowledge base. In the past, familiarizing with the data structure of the Fisheries legacy sources has proven to be a burden and delay for the development work.

- Design and implementation of software procedures that triples (i.e. produces RDF triples) the content of the Fisheries legacy sources. This task capitalizes the output of the web services operations, and produce RDF content that feeds the system knowledge base.

- The updated version of the FSDAS web client including all the fixes suggested by the users' feedback, and reported in [D7.7.2].

Table of Contents

List of Figures

List of Tables

1   Introduction

This document describes the work undertaken to update FSDAS application to the third and final prototype. After a major restyling of the application, from the first to the second release [D7.6.2], we focused on compelling as much as possible with the feedback provided by the end users and collected in [D7.7.2].

The work has been scoped around the main areas showing shortcomings highlighted by the usability test:

1.   Information retrieval process,

2.   Graphic Interface,

3.   Confidence in the validity of the returned result set,

4.   Help and support, and

5.   Terminology.

In the next section we present the list of the upgrades between FSDAS v2 and v3. All the interventions needed to improve from FSDAS v2 to v3 are listed in ANNEX3.

1.1   Differences to the first FSDAS Prototype

The evolution of the FSDAS prototype from the previous version is significant especially for the aspects concerning the maintenance of the FSDAS knowledge base. The realization of software procedures to systematically triple the content of "Fisheries and Aquatic Information and Statistics Service" (FIES) opens up to further development. If in the future more legacy data sources are identified to be processed, more RDF content will increase the current knowledge base.

The work of development included the following targeted objectives:

- The second prototype required the developers to be familiar with the FIES legacy data structure.

  The third prototype overcomes this limitation by wrapping the complexity of navigating the data structures within web services operations designed and implemented by Fisheries technicians. The web services expose only the data envisaged to be of interest for this version of FSDAS and can be broadened in the future.

- The second prototype had no integrated procedure to populate the system knowledge base.

  In contrast, the current version of the prototype is using a software java library (API) embedding the business logics to connect to Fisheries web services and generate the RDF triples according the ontology models delivered in [D7.2.4] and available online at: http://www.fao.org/aims/neon.jsp. If more web service operations (i.e. new data output) will be available then the java library can include new methods that

generate other RDF triples. In the same way if the ontology models evolve, the generation of the RDF triples is managed only in the API.

- The data being part FSDAS v2 knowledge base belong only to one source. No form of data aggregation from multiple sources was featured.

  The third prototype required the creation of ontology models about fish capture statistics from two different organizations: FAO, and Northwest Atlantic Fishery Organization[1] (NAFO). The two systems share only some of the dimensions along which a record of fish capture is composed; for them the ontology design could be reused, otherwise it required the definition of more concepts and relationships.

- The second prototype was found with some limiting factors in its features, when tested by end-users of Fisheries division.

  For the third prototype we focused on the needs for the users from the Fisheries division and thus expect better usefulness of FSDAS for the subject experts.

It is foreseen that if FSDAS will be adopted and used on the regular basis in FIES, other enhancements can be operated on future versions. This document goes together with the software developed for this prototype as a part of Task 7.6.

1.2 Structure of the document

In this document we first present in Section 2 the main goals and requirements of the FSDAS, along with the data sources used. Section 3 is devoted to explain the creation of web services that expose data of fishery domains to be included in the FSDAS system knowledge base, and wrapping the complexity of querying the legacy data structures holding them. Section 4 reports about data of the newly included fish capture domain and aquatic resource observation, together with their conceptualization by ontological models. Section 5 presents how the web services are used in a systematic way to populate the FSDAS system knowledge base with data from legacy data sources in the Fisheries division. Section 6 presents all of the enhancements and improvements achieved on the web client of the FSDAS application. After some conclusions and the list of references (Sections 7 and 8), we include as Annexes the installation guidelines and a brief user manual of the graphical user interface of the application.

---

[1] see http://www.nafo.int/.

2   Goals of FSDAS

FSDAS is an ontology-driven Fisheries Stock Depletion Assessment System. It embodies a knowledge base holding relevant information about fish stocks and associated knowledge about other types of entities. This knowledge is of two different types:

- Taxonomic knowledge stemming from multiple ontologies reengineered from the Reference Table Management System (RTMS) database of taxonomic terms.

- Factual information about many different entities stemming from XML fact sheets published by the FAO fishery department.

By the second iteration of implementation for FSDAS [D7.6.2] we reached a status of improved feature-set from the first prototype of FSDAS [D7.6.1] in a number of dimensions, e.g.

- Flexible query answering approach

- Access to real data

- Combination of keyword search and formal query mechanism

- Web-based user interface

- Employment of multiple, networked ontologies

This last iteration is devoted to make the above feature-set as robust as possible, driven by the feedback from the end users who tested FSDAS v2.

The original requirements document [D7.1.1] and the original design document [D7.5.1] for FSDAS system, define a number of use cases and requirements to inform the design and development activities. Some of them have been implemented in the first prototype, which has been evaluated in [D7.7.1]. This evaluation mainly points at the weaknesses in the presentation of the functionality and of the general look and feel. With the evaluation of the second prototype [D7.7.2] the focus was moved on the user study, to investigate whether and where the application needed to be improved with more/robust features and usability matters. These insights were considered when driving the development for the last iteration of FSDAS to the achievement presented in the introduction, and detailed in the section from 3 to **Error! Reference source not found.**.

2.1   Target User Group

The users of the FSDAS application will mainly be officers in FAO's Fisheries and Aquaculture Department FI (including Fishery Policy and Planning Division, Fisheries Resources Division, Fishery industry Division, as well as the Fishery Information, Data and Statistics Unit) attached to FAO headquarters and regional and sub-regional offices around the world. In this sense, the user classes are all fisheries experts within some domain of fisheries: Fisheries Scientists, Fisheries Managers, Marine Biologists, Oceanographers, Fisheries economists, Fisheries legal experts, Fisheries engineers, Fisheries policy makers.

Currently, FAO fisheries experts producing fisheries and stock assessments are forced to access numerous different information systems in order to locate the data they need to create their reports. This seriously hampers, both, searching for data and citing it. In addition, the necessity of searching through numerous systems means that reports may possibly neglect some data source one year, and use it the next, creating possible inconsistencies across years.

2.2 Use cases

The use-cases included in the third version of FSDAS have been maintained the same as for FSDAS v2. By the realization of the second application release, a process of use case elicitation was undertaken as a consequence of a development process more focused on few specialized cases, in favour of the application redeployed in a web environment.

The minimum set of use-cases to include, such that FSDAS was faithful to its characteristics given by the initial core objectives described in [D.7.1.1], is reported in Table 1. The plan for this development iteration has been to ensure improvements, for the existing features supporting the use cases, and to include those features that are still required to support the remaining use cases.

Table 1: Use Cases representative of the functionalities included in FSDAS v3.

Use Case	Comment	Priority		Second Prototype
		High	Low	
Search ontological resource in ontology	Search for ontology content as data value e.g.: codes, names, etc	☺		Functionality exists, but low coverage on all fishery domains is provided.
Search for related ontological resources	For a resource display the values from object properties and datatype properties e.g. "Ostrica" returns "ostrica gigas, ostrica viriginica, etc. and also from commodities "oysters fresh, oysters smoked, oysters canned. In addition it should be possible to see at least the links/titles of underlying data sources linked to such concepts	☺		Functionality exists, but needs to be made more robust.
Browse Taxonomy	Tree view; It is possible to go from ostrica gigas up to ostrica,	☺		Already available
Query Composition	It is possible to build a query based on the ontologies, and that the query choices are a function of the concepts and properties	☺		Functionality exists, needs to be improved in the interaction and presentation
Configurable Query Composition	There should be a configuration file/template so an admin could customise the query screen for a given ontology set	☺		Already available
Query for Data Source related	Retrieve a data source (i.e. factsheets) by querying the system	☺		Functionality exists, needs to be improved in presentation

Use Case	Comment	Priority		Second Prototype
		High	Low	
to individual	with elements from the loaded ontologies			
Visualize query results	Request info on abundance level of aquatic resources for which exploitation status is "fully exploited", get list of such aquatic resources together with abundance level. Also get links to data source (i.e. factsheets)	☺		Does not exist
Visualize Data Source related to individual		☺		Functionality exists, needs to be improved in presentation
Refine query	Means it is possible to amend elements from last query	☺		Already available
View ontological resource annotation	Annotations already attached to ontology	☺		Already available
View Data Source Annotation	Metadata for a factsheet, you would see it along with the URL		☺	Does not exist.

The prioritization of most mentioned use-cases is "high", representing their core value, and refers to a ranking system we started since [D7.5.1], and maintained for [D7.6.2] and [D7.7.2]. A number of other use cases were dropped from the expected functionality, e.g. features representing user management and collateral features such as sending search results attached to email or maintaining a set of favourite searches. These features were dropped in favour of more knowledge related functionality which demonstrates the power of ontologies for the fishery domain.

2.3 Sources of Information

Every day work of researchers and policy makers on fisheries, and stocks in particular, is based on a number of specialized sources of information (a comprehensive list of which was gathered and analyzed in deliverable [D7.1.2]) which always include numeric data (i.e., statistical data, on aspects such as fish catch, production, fisheries commodities etc.) and textual data such as reports, fact sheets and white papers. For this reason, two data sets representative of these two different sources of information were chosen as a basis for the FSDAS: the set of ontologies reengineered from the reference tables used to index time series (Section 2.3.1) and the fact sheets on fisheries produced by FIES, the "Fisheries and Aquatic Information and Statistics Service" within the Fisheries Department FI (Section 2.3.2).

This set of ontologies is the cornerstone of FSDAS building the network of ontologies that allows extracting, analyzing, and aggregating information for a fish stock assessment system. The conceptualizations are used to represent actual fisheries data, such as the geographical regions, the habitats actual fish stocks live in and the vessel and gear types that harvest them. FSDAS makes this information available, which is originally stored in a large number of XML files – the so called FIGIS fact sheets (cf. Section 2.3.2).

2.3.1 Reference data for FAO time series on fisheries

FIES collates time series concerning various aspects from fisheries. Data is provided by member countries and is then stored and published according to classification systems used to reference each piece of statistical data.

A time series is a sequence of statistical observations which are ordered in time and/or space. FIES collects observations about captures, aquaculture production, catches, fleets, trade of commodities, and consumption [FISTAT]. Any piece of statistical data is referenced by the following dimensions: time (in years), space (land and/or water areas), and the variable representing the observed object (e.g., biological species). For example, we can have statistical data about the catch of a given species in a given water area over a certain time span. In the case of statistics concerning trade, also the "trade flow" (import/export) is included. All statistics collected by FIES are available on the web and accessible by means of an online query panel.[2]

Reference data has been reengineered into ontologies in order to exploit the possibilities of semantic technologies. This work, carried on in T7.2, resulted in a network of OWL ontologies publicly available[3] and documented in two project deliverables; the first version of the network was documented in [D7.2.3], while the second version is reported in [D7.2.4]. The second network of ontologies includes ontologies populated with reference data, and with data of difference source. At the time of writing, the network covers the following domains: water areas (FAO divisions for statistical reporting, large marine ecosystems, exclusive economic zones), land areas (by reusing the FAO geopolitical ontology[4]), aquatic species (according to the taxonomic classification adopted in FAO, and according to the ISSCAAP grouping of species), classifications of vessel and gear types, aquatic resources, and statistical data about catch. It also includes two thesauri (ASFA[5] and AGROVOC[6]) and several sets of links between those ontologies,

2.3.2 Fact Sheets on Fisheries

FAO also generates a number of fact sheets about various subjects in fisheries. The fact sheets contents[7] is generated by experts and published electronically as XML documents according to a comprehensive XML schema [FSschema]. In this way, FAO makes available a large amount of information about fisheries, aquaculture and related subjects, including fishing techniques, fishing areas, fisheries and aquaculture country profiles. Fact sheets are grouped by *domains* (e.g., Cultured species, Fishing equipment, Fishery, Gear type), each corresponding to an element under the root FIGISdoc, the root of any fact sheet (XML

---

[2] See http://www.fao.org/fishery/statistics/en. For an in-depth description of time series and referenced data, see [D7.2.3].

[3] See http://www.fao.org/aims/neon.jsp

[4] For more information, see http://www.fao.org/countryprofiles/geoinfo.asp?lang=en.

[5] See http://www4.fao.org/asfa/asfa.htm.

[6] See http://aims.fao.org/website/AGROVOC-Thesaurus/sub.

[7] FAO fisheries fact sheets are published on http://www.fao.org/fishery/factsheets/en. They are the result of a complex procedure to make the content provided by the experts available in a format suitable for publication, after integration of information sorted in other systems (e.g. geographical maps).

document). Domains are fully specified by means of nested elements. Each element includes a description meant for human use.

The schema makes use of existing standard element sets such as Dublin Core [DC], Extended Dublin Core [EDC], AGMES [AGMES] and AIDA [AIDA]. It also incorporates wherever possible existing classification schemes (such as ISO standards for countries, currencies, languages, and other fisheries-related international classification schemes) most of which are stored in the reference tables.

It is important to note that the schema was conceived as a means for editors to create structured documentation, and as such was not created based on a relational or ontological model, but was rather organized following hierarchical document formatting conventions. A dictionary of the elements used in the schema is available online [FSdic].

3 Fisheries Web Services

3.1 Flattening Factsheet Data Structure

One of the concluding remarks presented at the end of the second iteration development [D7.6.2], is reported in the following quotation:

> *"[…] the FSDAS second prototype shows some shortcomings due to delays experienced during the development. […] For instance the generation of the knowledge base could have included other data repositories, but the time dedicate to the analysis and understanding of the fishery XSD data schema and Fishery naming convention, as well as coding systems, took longer than expected. This unexpected issue stole room for investigation in other sources of data."*

The process of populating the FSDAS knowledge base, with data coming from the species XML factsheets (cf. 2.3.2), consisted of:

Step 1.   XSL transformation of the XML factsheet source,

Step 2.   Further intervention for articulated section of the XML structure difficult to process with XSLT:

    a. Identify the relationship between different entities, e.g. an aquatic resource and a species via a key/foreign key relationship,

    b. Matching (non-key) attribute values, e.g. the scientific name the associations are first extracted literally with the XSLT style sheet,

    c. Consolidate and transform the correspondences above into real associations between objects,

Step 3.   Generation of RDF triples according to the fisheries domain ontologies.

The first two steps required knowledge built over a constant and protracted need for input from fishery experts, especially the expert of the XML schema. This created a dependency in the process which translated in a delay and a downscaling of the number of system features finally delivered.

For the third iteration of development we relaxed this dependency. We assured that the knowledge required for parsing the structure of an XML factsheet, was isolated and captured in a referencing mechanism supporting the first two steps in the list above. In other words, we hid the complexity of the factsheet structure to the FSDAS developers. The result is similar to having an array of factsheet data instead of a "tree" organization of data, typical of an XML document; the name of this operation was hence named "flattener". The logic to flatten down the XML factsheet structure, and to retrieve the data by a referencing mechanism, was coded into software operations exposed via public web services described in a WSDL file.

Providing such a support was required not only because new factsheets have been included as source for new RDF data, but also because those factsheets considered for

FSDAS v2 needed to be reprocessed for more data extraction. Moreover setting up this approach for feeding data from some of the Fisheries domains, was considered generally valuable for FIES, and responding to other software applications requirements.

3.2 Web Service Domains and Operations

To identify the operations to include as part of the flattener web service, we followed the following pragmatic approach: since the goal was to reduce as much as possible the dependency of the developers from the input of experts in the Fisheries division, we designed at first operations for extracting those data already part of FSDAS v2 (see Table 2). After that, we analyzed the evaluation document of FSDAS v2 (i.e. D7.7.2), in search for which additional knowledge was required to be part of the data collection for FSDAS v3. This pointed us to the new data target of the triplification process, and hence the XML factsheets set to be added to the initial collection.

In Table 2 and Table 3 there are the complete lists of operations that have been designed (cf. 3.3) and implemented (cf. 3.4), and it is indicated to which domain of fishery they belong. Factsheets of *aquatic species domain*[8] is covered by a collection of XML documents (presented as HTML pages) each one aggregating data on a single aquatic species. This class of factsheets can be assumed as the "identification card" for an aquatic species where we can read what is its scientific name and synonyms, where and at which depth does it live in the sea, what is its biological habitat, and more. The *Aquatic resource observation domain*[9] is covered by a collection of XML documents (presented as HTML pages) each one reporting on the status-quo of an aquatic resources (i.e. one or more aquatic species) living in specific water areas at a certain point in time. The information that this class of factsheets holds is possibly the most significant for carrying out stock assessment, since they already are the result of monitoring activities carried out by a different groups in more then one organization, but lacking explicit data cross referencing, provenance and other relationships. Aquatic resource observation is the new domain now covered by FSDAS v3, by including extracted data into the system knowledge base (cf. 5).

Table 2: List of operations for the domain of "Aquatic Species" factsheets[10]

Operation Name	Input	Output	XPath
get3AlphaCode	sID	3AlphaCode	fi:figisdoc/fi:aqspecies/fi:aqsIDent/fi:fao3alphacode
getMaxBathymetryValue	sID	Batimetry Value	fi:figisdoc/fi:aqspecies/fi:aqspeciesfeature/fi:habitatbio/fi:depthbehav/fi:bathymetry/fi:Max
getMinBathymetryValue	sID	Batimetry Value	fi:figisdoc/fi:aqspecies/fi:aqspeciesfeature/fi:habitatbio/fi:depthbehav/fi:bathymetry/fi:Min
getScientificName	sID	ScientificName	fi:figisdoc/fi:aqspecies/fi:aqsIDent/fi:sciname/fi:genus + fi:figisdoc/fi:aqspecies/fi:aqsIDent/fi:sciname/fi:species

---

[8] See http://www.fao.org/fi/website/FISearch.do?dom=species&lang=en.

[9] See http://firms.fao.org/firms/search/en.

[10] The input for the operations is the ID for an aquatic species (sID).

NeOn

Table 3: List of operations for the domain of "Aquatic Resource Observations" factsheets[11]

Operation Name	Input	Output	XPath
listFishingAreas	(oID, arID)	Collection<fishingAreaID(s)>	fi:figisdoc/fi:aqres/fi:aqresident/fi:waterarealist/ fi:waterarearef/fi:figisid/text()
listSpecies	(oID, arID)	Collection<speciesID(s)>	fi:figisdoc/fi:aqres/fi:aqresident/fi:specieslist/fi: speciesref/fi:figisid/text()
listVerticalDistribution	(oID, arID)	Collection<v-Distributions>	fi:figisdoc/fi:aqres/fi:habitatbio/fi:verticaldist/@ Value
listHorizontalDistribution	(oID, arID)	Collection<h-distributions>	fi:figisdoc/fi:aqres/fi:habitatbio/fi:horizontaldist/ @Value
listClimaticZone	(oID, arID)	Collection<climaticZones>	fi:figisdoc/fi:aqres/fi:habitatbio/fi:climaticzone/ @Value
listGeoForm	(oID, arID)	Collection<geoForms>	fi:figisdoc/fi:aqres/fi:habitatbio/fi:geoform/@Value
listBottomTypes	(oID, arID)	Collection<bottomTypes(s)>	fi:figisdoc/fi:aqres/fi:habitatbio/fi:bottomtype/@ Value
listExploitationStates	(oID, arID)	Collection<exploitationStates>	fi:figisdoc/fi:aqres/fi:aqresstatetrend/fi:exploitst ate/@Value
listStdExploitationRates	(oID, arID)	Collection<std-exploitation-rates>	fi:figisdoc/fi:aqres/fi:aqresstatetrend/fi:exploitra te/fi:standardexploitrate[1]/@Value
listStdAbundanceLevels	(oID, arID)	Collection<std-abundance-level>	fi:figisdoc/fi:aqres/fi:aqresstatetrend/fi:abunda ncelevel/fi:standardabundancelevel/@Value
listDepths	(oID, arID)	Collection<detpth-zones>	fi:figisdoc/fi:aqres/fi:habitatbio/fi:depthzone/@ Value
getReportYear	(oID, arID)	Year	fi:figisdoc/fi:aqres/fi:aqresident/fi:reportingyear
getReferenceYear	(oID, arID)	Year	fi:figisdoc/fi:aqres/fi:aqresident/fi:referenceyea r

---

[11] The input for the operations is the ID for an observation (oID) together with the ID for an aquatic resource (arID).

3.3 Design of the Web Service Operations

The main goal for having in place a referencing mechanism that retrieves data from the factsheets was to relax the dependency of FSDAS development work from the availability of input from experts of FIES. In other words, we can avoid referring to a person source of the required information, but rather to a system provided with similar knowledge. For this reason, a number of software operations (see Table 2 and Table 3) were designed and named after the specific data/data-collection they output, and then exposed via SOAP web service for an easier integration with multiple software development scenarios in FIES.

The design consisted of the following steps:

1. Identification of single-valued data in the set of information collected in the factsheet,

2. Identification of an ID value to reference the XML factsheet document in the collection,

3. Naming the operation with respect to the data kind identified at step 1,

4. Associating the XPath to the data value for each operation.

The first step assured that the value retrieved could be directly used to assert an RDF statement safely with both an owl:ObjectProperty or an owl:DataProperty. We chose to initially focus on single-valued data for priority reasons, since long-text-value would need more operation of post-processing to extract significant information in the scope developing FSDAS v3. Moreover we restricted our choice among all and only data already part of the system knowledge base.

The second step is crucial to assure the compatibility with other information systems in FIES; in other words we wanted to be compliant and coherent with previous design choices made for existing data dissemination application like the FIGIS web portal. For instance a factsheet on aquatic species is retrievable online via an URL that has this pattern:

http://www.fao.org/fishery/species/2011/en;

We analyzed the structure of the URL to isolate the number (i.e. 2011) that uniquely identifies the species in the FIGIS system and adopted it as input for the operations.

The third step is basically functional for human consumption; the naming of the operation will carry the information on which data it is returning as output. This is following the conventions and best practices of JAVA programming language. The output of an operation can be either a single value, or a collection of single values (see Table 3), and is determined by the schema of the XML factsheets.

The fourth and last step is where the fundamental part supporting the motivation of having web services, takes place. The XML schema expert that was constantly the proxy of communication between FAO and other partners participating in developing FSDAS v3 will input once and for all the path to retrieve a specific data kind. When doing so he will employ all his background knowledge of XML schema and factsheet structure, finally materialized in the association made between operation and XPath. The schema expert is as well involved in identifying which are the single-valued data available in the two factsheet domains.

The last step in designing the web service operation clarifies why this approach is generally valuable for Fisheries division and FSDAS; any evolution at the level of the factsheet XML

schema or template structure, will be handled by one person one time, without the need of notifying any consumer of the web service. This is true as long as the web service operations will stay the same.

3.3.1   Operations of Aquatic Species Domain

Operations design followed a bottom-up approach, starting from those data already included in the knowledge base of FSDAS v2, and extracted with a cumbersome workflow requiring hybrid approach of XSL transformation and logics for matching key/foreign-key attributes with other species identifiers.

Table 2 reports the result of this design phase in terms of named operations, definition of input and output, and the XPath to the data value inside the XML factsheet.

The input for all of the operation will be a species identifier (sID) provided by an operation external to this set. The identifier we mentioned in the previous section is sufficient to retrieve an XML factsheet resource, when is prefixed with internet domain of FIGIS system, and trailed with the language chosen for the presentation. After the XML file is retrieved, the XPath associated with the operation allows one to reach and extract the data as value in return.

3.3.1.1 How to use the web service

To start to get an insight on how the web service can support the developers when populating the knowledge base, we present an example based on the operations in the domain of aquatic species. More detailed explanations on usage and integration of web service with the procedure to populate the knowledge base are presented in section 5.

Figure 1 shows a snapshot taken from the web page retrievable at this address:

http://www.fao.org/fishery/species/2011/en.

This page is a rendering of the content of the respective XML factsheet[12] as an HTML page, in favour of human consumption. As we said, and it is visible in the picture, an XML file collects a number of information, but not all are relevant to our scope.

---

[12] See http://www.fao.org/fishery/xml/species/2011/en.

**Figure 1: Snapshot of the fact sheet for the species <u>Shortfin mako</u> at
<u>http://www.fao.org/fishery/species/2011/en</u>.**

Applying the logics of extraction presented in Section 3.3.1 to the XML source available at this URL: <u>http://www.fao.org/fishery/xml/species/2011/en</u>[13],

and for all of the operations in Table 2, a developer is provided with the support to invoke any of the operations and to retrieve the data as reported in

Table **4**. The developer will apply no effort in locating the XML source or updating his knowledge on the possible evolution of the XML structure.

One typical RDF statement that can be serialized when these operations are programmatically invoked within the knowledge base population software procedure can be the following:

```
#PREFIX

species:http://www.fao.org/fishery/species/

#STATEMENT

species:SMA species:hasScientificName "Isurus oxyrinchus"
```

---

[13] Note that this URL points to the XML source that is used to render the HTML page with XSL transformation

Table 4: Example of input and output obtained for the species <u>Shortfin mako</u> (2011[14]).

Operation Name	Input	Output
get3AlphaCode	2011	SMA
getMaxBathymetryValue	2011	152
getMinBathymetryValue	2011	0
getScientificName	2011	Isurus oxyrinchus

3.3.2 Operations of the Aquatic Resource Observation Domain

An aquatic resource can be one or a set of aquatic species, generically referred as resource. An observation is a terminology borrowed from statistical context to mean the event of measurement of typical dimensions characterizing a resource. A factsheet for aquatic resource observation is a report of such measurement on the observed dimensions. The dimensions of interest to us can be read in the column headed "output" in Table 3, for instance the exploitation or conservation status, the level of abundance, all refer to the subject aquatic resource. Since FIES collects records of observations in time, an observation XML file is uniquely identified by two key values, namely the year, when the measurements were detected, and the resource identifier (i.e. a code).

The application of the design procedure to the domain of aquatic resource observation led to the list of operations presented in Table 3. The Aquatic Resource Observation (ARO) domain is introduced in FSDAS v3, and the specifications for the web service were based on the feedback gathered from the evaluation of the second release of this system.

The first step in the design process was carried out with the support of the XML schema expert. Among all the data contained in an ARO XML factsheet, we identified only the single valued ones, and that are relevant to our scope and aim. Most of these data are enumerated xml data type, for instance the "exploitation state" is a kind of data that can assume only the following values: *OverExploited* or *UnderExploited*; hence envisaging the output of such operations was trivial decision.

Identifying an ID for an XML factsheet such to uniquely reference it in the collection of all observations files, considers actually a pair of IDs, one for the observation and the other for the aquatic resource. This choice is supported by how an observation factsheet is retrievable over the internet through the FIRMS web portal. For example, the fact sheet reporting an observation acquired in 2005 about the aquatic resource with ID "874" can be found at the following URL:

http://firms.fao.org/firms/resource/10094/874/en

where the numbers "10094" and "874" are the codes for the observation and the aquatic resource, respectively. Adopting this referencing mechanism already in place in FIES we

---

[14] See http://www.fao.org/fishery/species/2011/en.

designed the input as the pair observation ID (oID), and aquatic resource ID (arID). The pair is retrieved with another operation external to this web service.

Naming the operations of the web service is the most trivial step since it is supported by practices of JAVA programming. Differently from the web service for aquatic species domain, the output of most of the operations in this domain is a collection of values; when this happens the operation name is prefixed with "list-", otherwise with "get-".

Once the list of operations name, input and output was set, the XML schema expert provided with the XPath to the data in the XML factsheet template.

3.4 Implementation of the Web Service Operations

The software is using a domain driven design approach. This means that the model of the domain is implemented as a set of java beans, representing the model. Jibx[15] is used for the conversion to and from xml, and XPath is used to pick the elements from the factsheets.

The diagram in Figure 2 shows the internal structure of the web service. Note that internally the flattener is using the service layer from the factsheet, called factsheet-services. This is the layer that provides the external operations to retrieve the IDs used as input for flattener operations (i.e. species-ID, observation-ID and aquatic-resource-ID).

The top level artefact, **flattener-web**, is the web-layer responsible for handling the web service requests, conversion of xml to java beans and vice versa. The **flattener-services-def** is the utility project of using the web service framework Axis2; it contains the service.xml, which is the definition of the services in Axis 2 terms. The **flattener-service** is the logical service layer for the flattener, and is not web service specific. The f**actsheet-service** is the logical service layer for the factsheets, and is not web service specific. The **flattener-domain** contains all the domain objects in the form of java beans, and the **factsheet-domain** contains all the domain objects in the form of java beans.
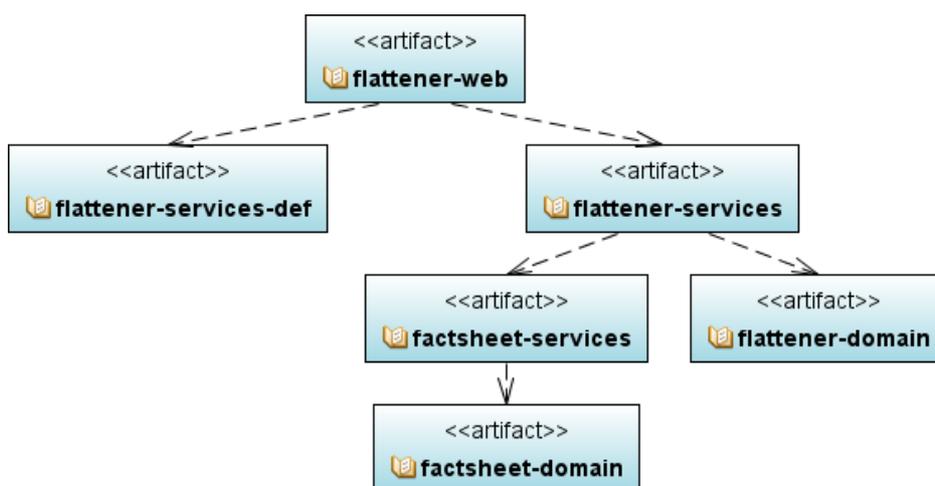
Figure 2: Component diagram of the web service.

---

[15] See http://jibx.sourceforge.net/.

At the time of writing, a WSDL based web service architecture is used, based on Axis2, but in the future this will be changed towards a REST-based architecture, complying with other change in the FIES information systems.

All software written for the flattener web service can be found in subversion on this location:

https://openfigis.svn.sourceforge.net/svnroot/openfigis/trunk/flattener/

4 Statistical Data

Beside the introduction of the web services, the innovative part of FSDAS v3 also concerns the data sets used, which now include aquatic resource observations and fish capture time series. Both these domains deal with records of data reported after a measurement activity, and both feed the statistical data domain. Both of them have been analyzed with the support of FIES experts and given a graph representation of the data record used to store them. From this initial draft schema a further analysis, still with support from FIES experts, led to a more semantically rich conceptualization of ontology models.

Having the ontological models of the two domains is functional to the process of tripling the sources of data identified for each of them. Namely, the aquatic resource observation factsheets (via the associated web service, cf section 3.3.2), and a CSV collection of fish capture records from FAO and NAFO organizations.

In the following we introduce briefly the two domains and the graph of the fields of their data record, and then a presentation of how they developed in coherent owl ontologies.

4.1 Capturing Knowledge of FAO and NAFO Time Series Domain

To promote data aggregation from different systems, this version of FSDAS included data about fish capture gathered from two organizations: the Northwest Atlantic Fishery Organization, and FAO. Both these organizations arrange and disseminate statistics on amount of fish captured along their pertaining water areas. In the scope of this work these data were sampled and distributed in a comma-separated-value format. Data from FAO were exported through a web client application available online (see footnote 2); data from NAFO were fetched from an SQL database dump distributed via email. The level of details of the two datasets was different, which required a first step of harmonization to reach a common catch record definition.
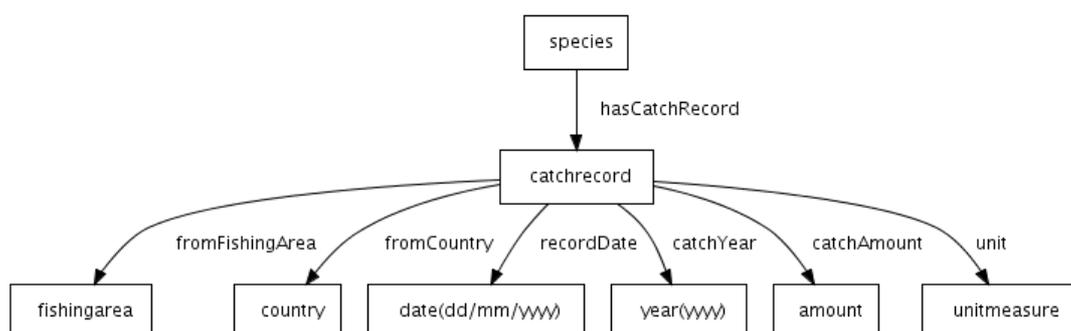
Figure 3: Graph representation of a catch record common to FAO and NAFO time series.

From this initial work of harmonization, a common catch record was defined, holding the dimensions visible in Figure 3. The picture shows a graph-like view of the catch record, highlighting the subject and object of the relations (arrows) holding among them. This structure captures knowledge of FIES experts in the domain of NAFO and FAO time series; as such it was submitted to the ontology engineers who were able to start drafting an ontology model, and further refining it with more support from FIES experts.

4.2 Capturing Knowledge of Aquatic Resource Observation Domain

Capturing the knowledge of Aquatic Resource Observation domain in a graph-like structure was handled with a process similar to that described in the previous section. The phase of domain analysis, i.e. to isolate the parameters of an observation record, coincided with the identification of the single-valued data at the time of defining the operations for the web service in the same domain (cf. Section 3.3.2). It is in fact visible the overlap of the record dimension with the output of the web service operations. This overlap is self explained if we think that the tripling process will populate the structure in Figure 4 taking the input from the operation in the web service.
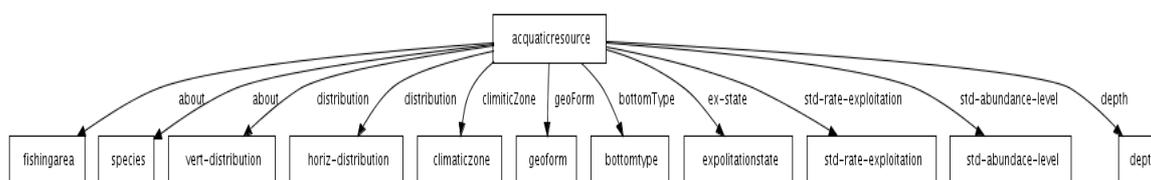

Figure 4: Graph representation of an aquatic resource observation record.

The structure in Figure 4 captures knowledge of FIES experts in the domain of aquatic resource observation; as such it was submitted to the ontology engineers who were able to start drafting an ontology model, and further refining it with more support from FIES experts.

4.3 Ontology Model for FAO and NAFO Time Series

The catch record ontology reuses the content design pattern that models observations, records, and statements of dynamic facts, with a specific temporal indexing[16], and the content design pattern that model localization relations and places[17]. Figure 5**Error!**

---

[16] See http://ontologydesignpatterns.org/cp/owl/observation.owl.

[17] See http://ontologydesignpatterns.org/cp/owl/place.owl.

Reference source not found. shows the ontology import graph for the catch record ontology (arrows represent owl:imports statements):
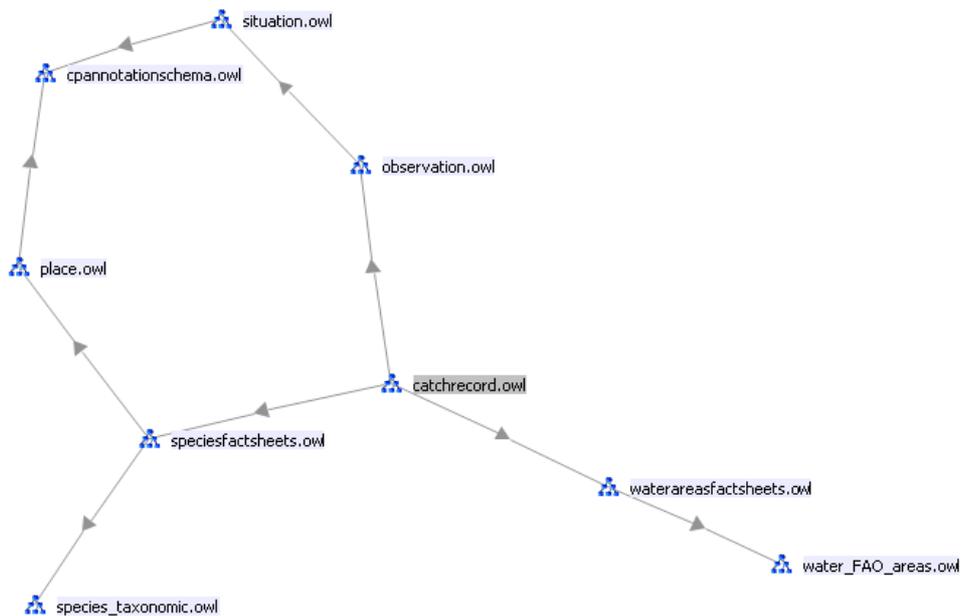

Figure 5: The import graph for the catch record ontology.

The taxonomy of classes for the catch record ontology is depicted in Figure 6.
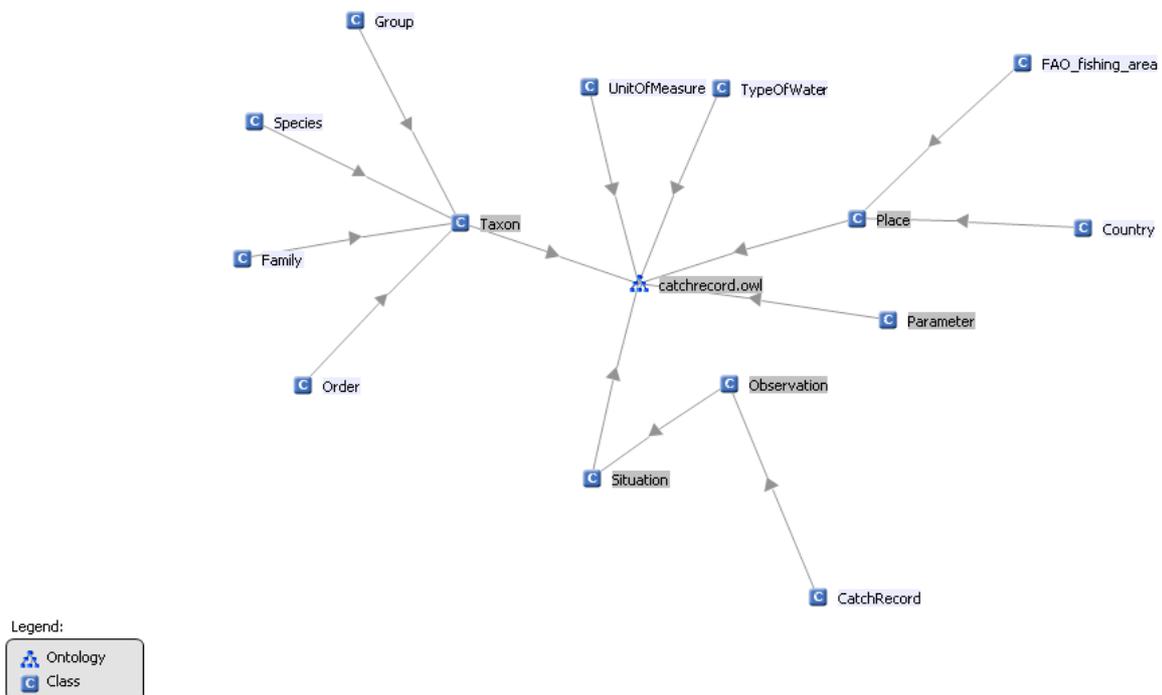
Figure 6: The class taxonomy of the catch record ontology.

Finally, the basic pattern from the catch record ontology (Figure 7) represents the graph of object properties holding for the CatchRecord class, which can be considered a domain-oriented pattern in itself:
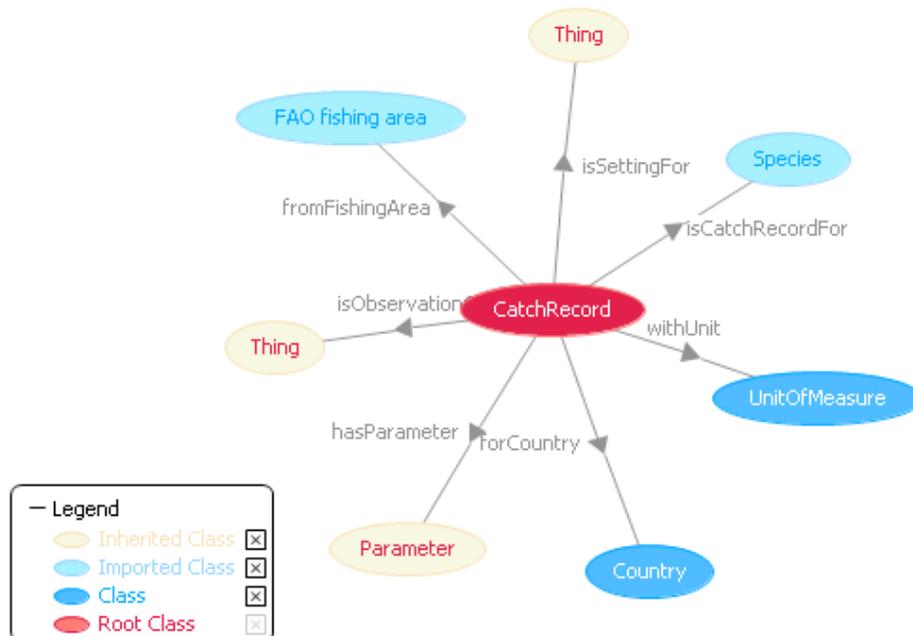
Figure 7: The catch record pattern.

The ontology also contains several datatype properties, the interested reader should refer to the OWL code at its URI for details.

4.4 Ontology Model for the Aquatic Resource Observation Domain

The aquatic resource observation ontology reuses again two content design patterns: one for modelling observations, records, statements of dynamic facts, with a specific temporal indexing[18], and one for modelling localization relations and places[19]. Figure 8 shows the ontology import graph for the aquatic resource observation ontology (arrows represent owl:imports statements):

---

[18] See http://ontologydesignpatterns.org/cp/owl/observation.owl.

[19] See http://ontologydesignpatterns.org/cp/owl/place.owl.
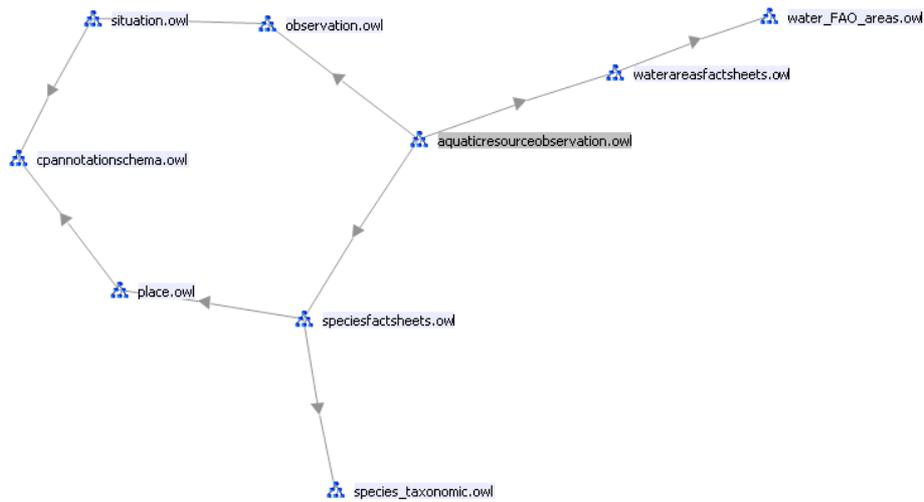
Figure 8: The import graph of the aquatic resource observation ontology.

The taxonomy of classes for the aquatic resource observation ontology is depicted in Figure 9:
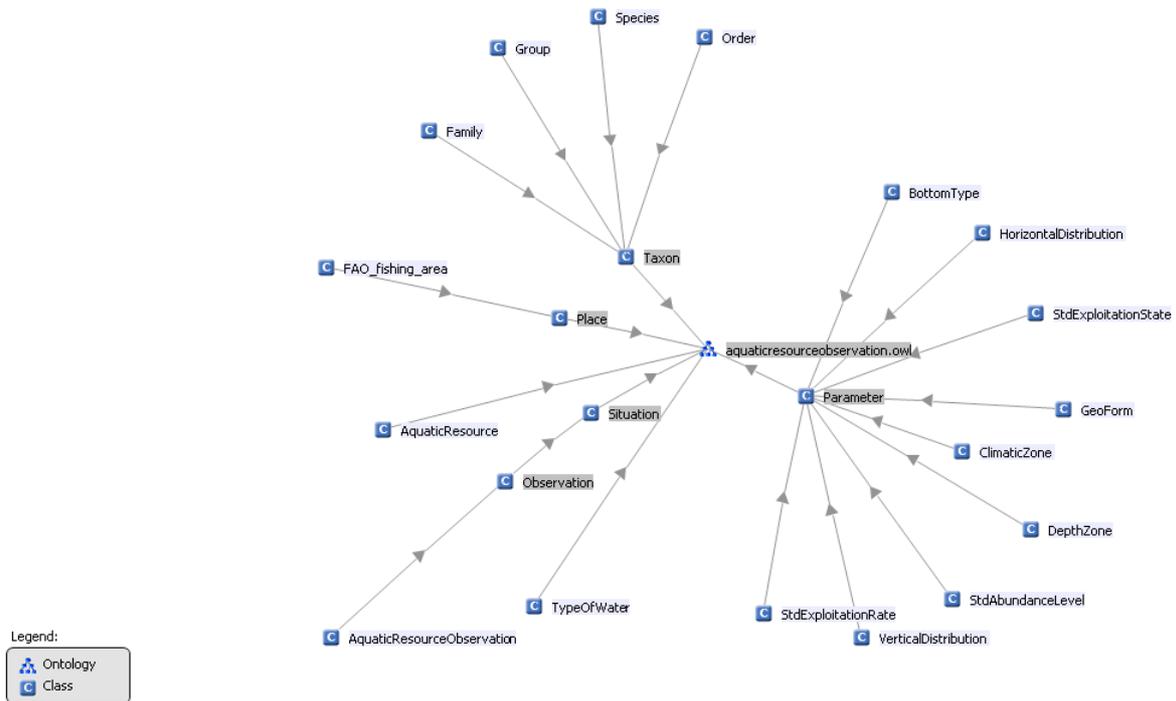
Figure 9: The class taxonomy of the aquatic resource observation ontology.

Finally, the basic pattern from the aquatic resource observation ontology (Figure 10) represents the graph of object properties holding for the AquaticResourceObservation class, which can be considered a domain-oriented pattern in itself:
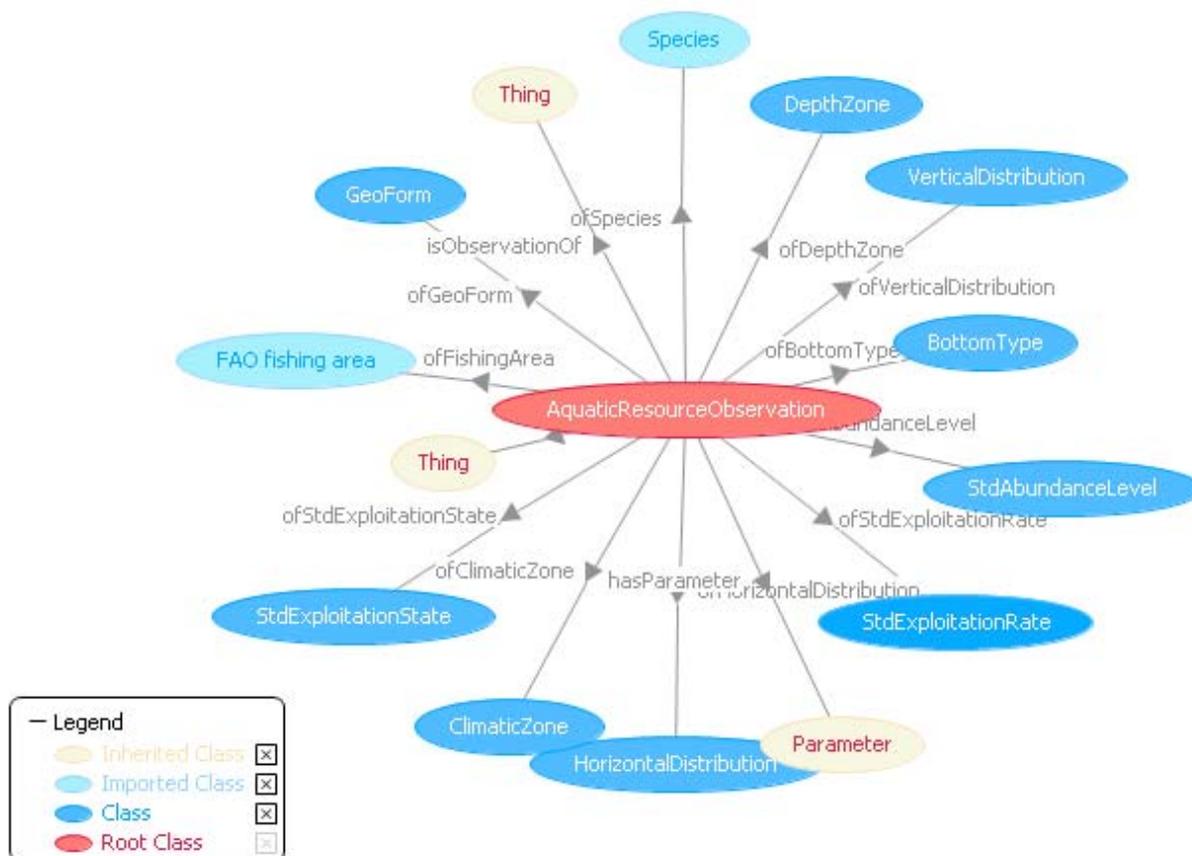
Figure 10: The aquatic resource observation pattern.

The ontology also contains several datatype properties, please refer to the OWL code at its URI for details.

5 Knowledge Base Population

5.1 Introduction

The FSDAS system is based on original FAO data sources (cfr. Section 2.3). In addition to taxonomic and terminological information coming from the Reference Table Management System (RTMS) reference data base a big number of XML fact sheets provide information about various (more dynamic) entities, e.g. about observations of aquatic resources, i.e. their development over time, or catch data.

The Knowledge base consists of the following sources:

- The network of ontologies representing taxonomic and terminological information about:
  - o Species Taxonomic
  - o Fishing areas
  - o Geopolitical Ontology
- Ontologies that interlink different entity types and also model the observations and time-series data
  - o Aquatic resource observations
  - o Catch record ontology
- A huge amount of time-related basic facts for catch data as collected by FAO and NAFO and is provided in the form of text files with comma separated values.
- Finally, even more basic facts about observations of aquatic resources and species are provided in the form of web services.

In this section we present two major improvements to support : (a) the access to the text-based and the web-service based data, and (b) the extraction of the data from these original sources. Both the procedures are preparatory to populate the knowledge base for FSDAS.

5.2 Architecture

The architecture of the *tripling* component is rather simple. It is a standalone Java application, which can be configured via some properties-files and which accesses the CSV files and web-services to generate raw RDF data which adheres to the terminology defined in the ontologies and can be deployed into the FSDAS system.

Since the original data does not change very frequently and only needs to be updated about once a year, this simple mechanism is sufficient to fulfil all requirements.
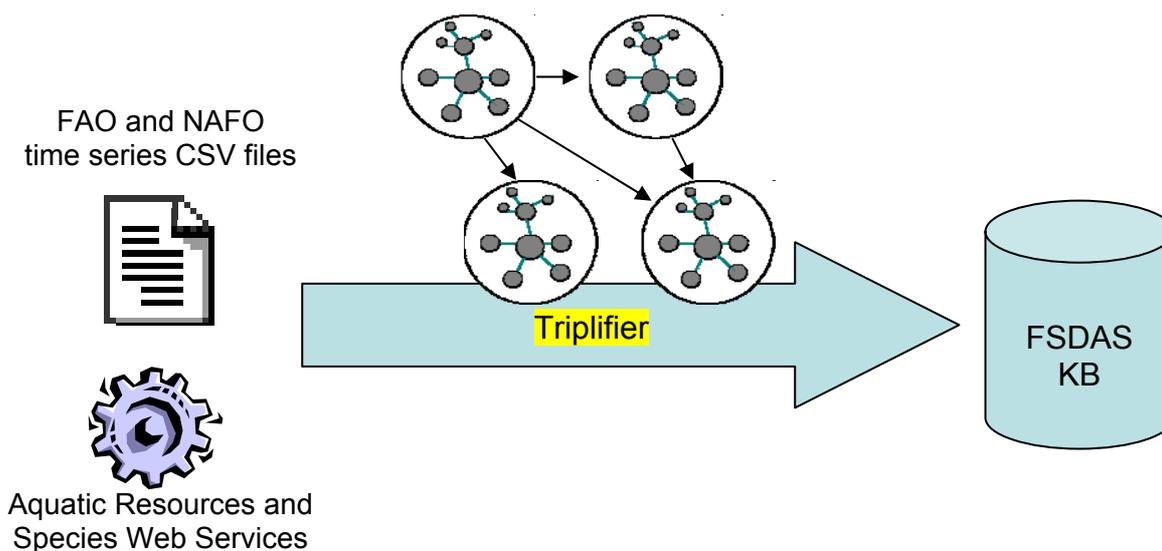
Figure 11: The high-level architecture of the *tripling* component.

5.3 The Population Workflow

The processing of the input is the same for all sources, with the exception of the NAFO time series files, which need some pre-processing in order to collect all needed information in one data structure. The main steps for the *tripling* are:

1. Open the document or the connection to the web service.

2. Iterate over all available records.

3. Translate each record into RDF statements using the vocabulary from the ontologies

4. Serialize the resulting RDF model into a file for deployment on the FSDAS server.

5.3.1 Accessing the Sources

The process of creating the basic triples for the knowledge base starts with accessing the sources. In case of the Web Services, this is easy since they are available on-line and simple SOAP requests are sufficient to connect to them and to get all needed information.

The initiation of the processing for the CSV files has to be manual be providing the needed files and by explicitly pointing the extraction procedure to the files.

5.3.2 Extracting Basic Information

Iterating over the FAO time series CSV file is easy, since it contains one record per row. The structure of the file looks like this:

Figure 12: Structure and sample of the FAO time series CSV file.

The CSV files for the NAFO time series have to be pre-processed in order to allow for easier tripling in later steps of the process. In the diagram below, we see the relationships between the different tables.
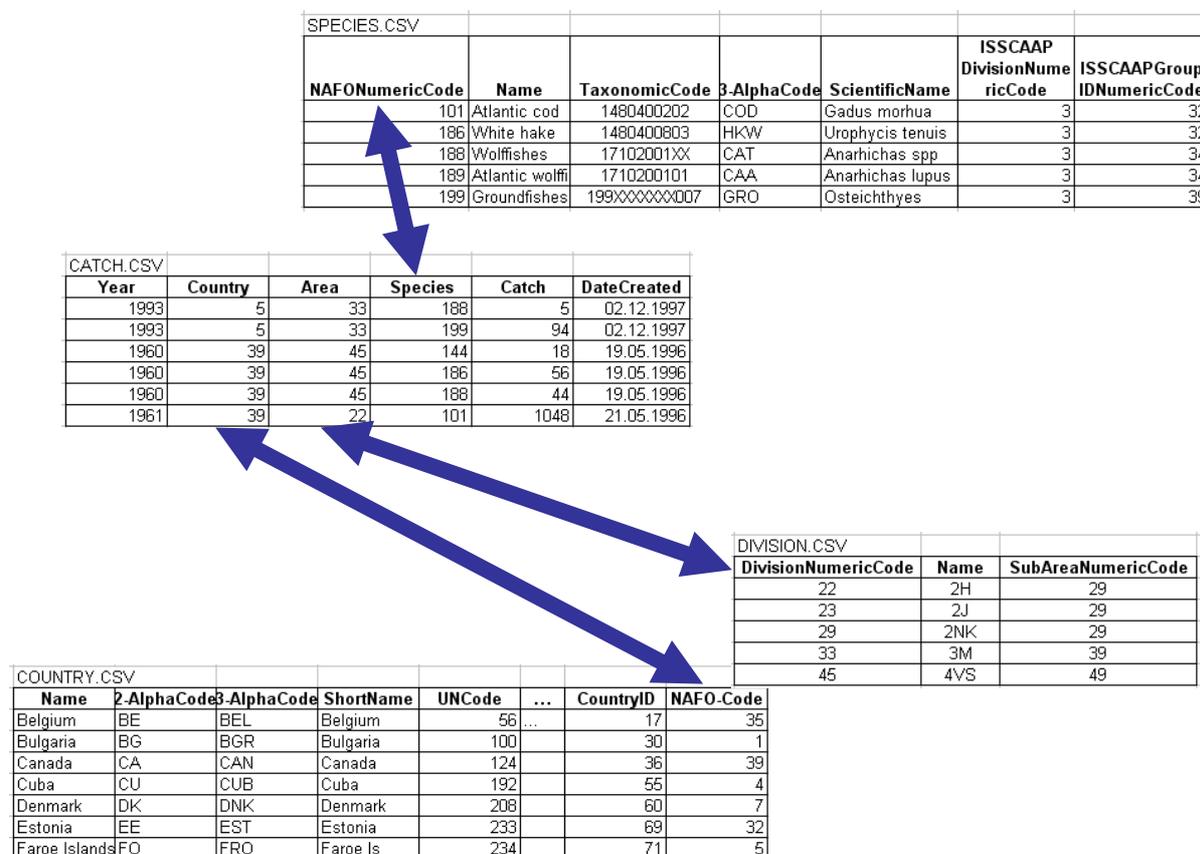

Figure 13: The relationship between the four NAFO CSV files.

The resulting table for the NAFO time series looks like this.

CATCH-PROCESSED.CSV					
Year	Country	Area	Species	Catch	DateCreated
1993	FRO	33	CAT	5	02.12.1997
1993	FRO	33	GRO	94	02.12.1997
1960	CAN	45	USK	18	19.05.1996
1960	CAN	45	HKW	56	19.05.1996
1960	CAN	45	CAT	44	19.05.1996
1961	CAN	22	COD	1048	21.05.1996

Figure 14: The pre-processed NAFO CSV file.

This structure is equivalent to the structure of the FAO time series data and, thus, can be processed with similar procedures like the FAO case.

Iterating over the "contents" of a web services is not possible in general. However, the services provided by the FAO fisheries department provide special operations to get all IDs for which they can provide more information. That means for iterating over all records of the web services we first request all relevant IDs and then iterate over this set to construct the records with all information the service provides for this ID.

5.3.3 Generating Triples

Since the ontologies have been constructed in such a way that their terminology and structure is compatible to the information that is contained in the data sources, the translation process from records to sets of RDF statements is relatively easy. However, there is one issue which complicates the matter drastically: In the Semantic Web world all entities have unique URIs to identify them, which means that the process for populating the FSDAS KB also needs to create RDF resources (in our case OWL individuals) for the different entities. The web services and CSV tables do define relationships between different entities, e.g. the web services relate aquatic resources and species via the key/foreign key relationship or the time series relate a catch record to the species or the country via matching (*non key*) attribute values like the 3-alpha code. These associations have to be properly represented in the RDF graph by translating them to useable object-property links between the related individuals.

This can be achieved in two ways:

- **Via naming conventions**: if the value used to refer to another individual is a the value of a key property and the naming scheme for the individuals use the key value as part of the URI in a canonical way, knowledge of the key value is sufficient to compute the actual URI for creating the appropriate OWL individual for establishing the right object property link.

- **Queries against the ontologies to retrieve the intended object URI for the RDF statements**: A more involved but also more general solution requires querying a knowledge base for the needed OWL individuals. Since the time series documents refer to species and countries via there 3-alpha code, which is not part of the naming scheme for species or countries, the proper URIs must be retrieved from the Taxonomic and Geopolitical ontology, respectively. A sample SPARQL query against the Geopolitical ontology to retrieve the proper URI for the country with 3-alpha code "FRO" would look like this:

```
PREFIX geo: http://www.fao.org/aims/geopolitical.owl#
PREFIX rdf: http://www.w3.org/1999/02/22-rdf-syntax-ns#
SELECT ?COUNTRY
WHERE {
    ?COUNTRY  rdf:type  geo:self_governing.
    ?COUNTRY  geo:codeISO3  "FRO"^^xsd:string.
}
```

This query will yield the intended URI for the actual country that should be referenced: <http://www.fao.org/aims/geopolitical.owl#Faroe_Islands>

Only when proper URIs have been identified correct and meaningful RDF triples can be generated. This step is crucial for a coherent and usable knowledge base.

5.3.4 Storing Results and Deploying the Knowledge Base

This process yields one RDF fact file for each source. The final step of the process is simply the import of this set of files into a single knowledge base where they are combined with the network of ontologies (cf. D.7.2.4).
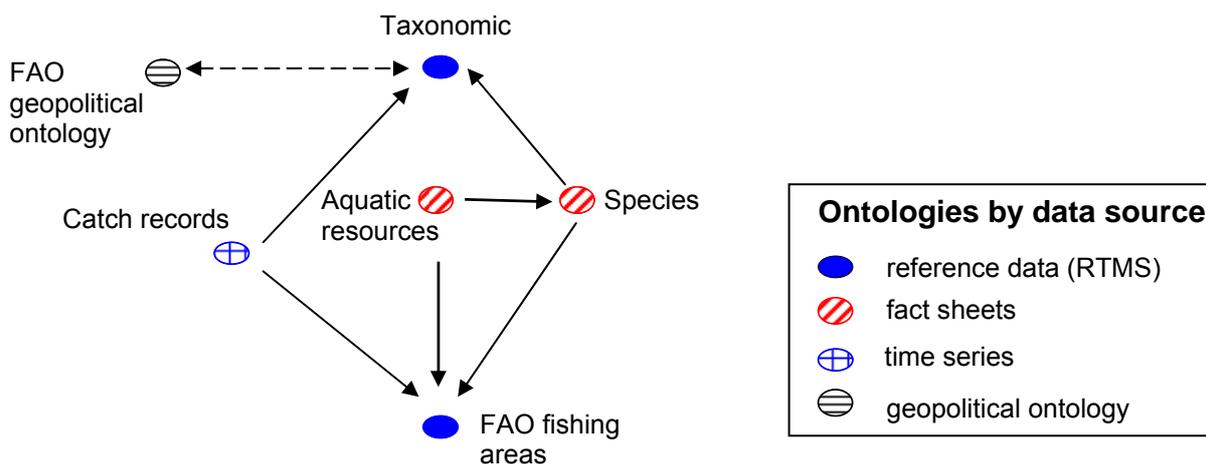
Figure 15: The runtime ontologies of the FSDAS [adapted from D7.2.4]

This KB is exposed to the FSDAS application via an API providing a SPARQL interface, which is used to populate the GUI.

6 FSDAS Web Client Update

Both versions of FSDAS previously developed, have been evaluated by representatives of the envisaged class of end-users (cf. 2.1) in FIES. The most striking result passing from FSDAS v1 to v2 was the change of the system architecture in favour of a web application (v2) from a desktop one (v1). The second evaluation by end-users [D7.7.2] reported a number of shortcomings around 5 main evaluation aspects:

1. Information retrieval process
2. Graphic Interface
3. Confidence in the validity of the returned result set
4. Help and support
5. Terminology

The feedback, gathered by a script-based usability test, carried out with talk aloud technique, was compiled in a structured format and submitted to developers of FSDAS web client and server. After a first analysis of the prototype limitations, which led to the definition of prioritizing and job scheduling, we propose a technical solution, for which Annex 3-6 of this deliverable present a tabular overview. In the remainder of this chapter we present the new features and improvements implemented into FSDAS v3.

6.1 Information retrieval process

This evaluation aspect is defined as the set of FSDAS capabilities involved in the process of fish stock assessment, from user interaction, to the implemented retrieval operations. Annex 3 lists all of the interventions needed to improve from FSDAS v2.

- FSDAS v2 had a keyword based search implemented as case sensitive, and depending on the charset.

  In the current version of the prototype, the indexing algorithm based on Ontobroker has been customized to eliminate dependencies on the keyword case, and the charset. More over the index includes more data, as it is build over an enlarged version of the knowledge base, containing more name translations and Latin names for the species. This ensures a richer set of results.

- The user of FSDAS v2 had to select at least one query template in the interface, before triggering the retrieval process. Otherwise only an error message was displayed.

  This constrain has been eliminated in FSDAS v3 checking automatically the template for which the user perform an input. In case the user triggers a query process accidentally with no templates filled in, the system shows a warning message reminding to input at least one valid value.

6.2 Graphical Interface

This evaluation aspect deals with the accessibility of the system functionality by the end users. The feedback gathered in [D7.7.2] is mostly related to the need to have a more coherent presentation of all the graphical user interface (GUI) elements, and to improve the layout components to make them better accessible and more intuitive. Annex 4 lists all of the proposed improvements compared to FSDAS v2.
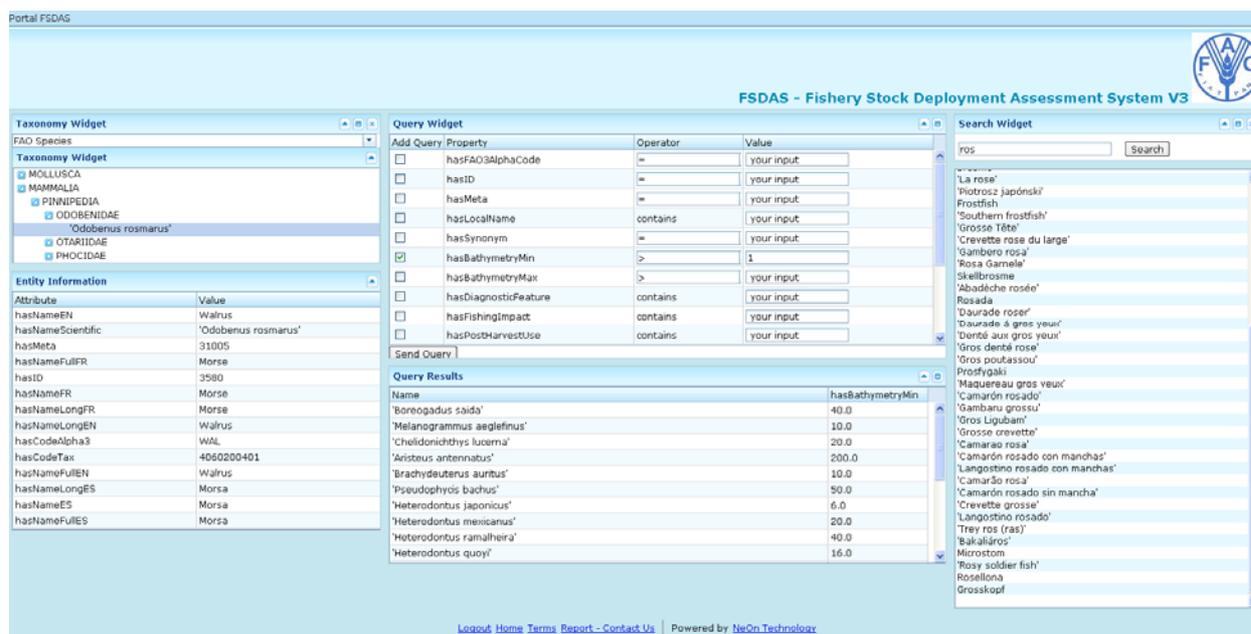

Figure 16: FSDAS v3 - Main GUI.

From a graphical point of view, results provided from the query or search are displayed now as a linkable image instead of a URI, because labels are more meaningful to the end-user.

- Users complained about the interface layout. They were expecting a similar layout and behaviour to the applications that they use in their daily business life.

  FSDAS v3 adopts a different layout system from version 2, which allows users to move GUI panels around, to better suit their preferences. Similarly to a windows-based desktop application each user can create his/her own panel positioning, and size when using the system. Technically, this layout is named "PortalLayout" and is also adopted by other web applications, e.g. iGoogle.

- The panels devoted to list items adopted un-coherent browsing methods. Some of them were provided with a *scrolling* feature, others used *pagination*.

  The panels of the GUI v3 are equipped with re-sizing buttons now, which let users increase their size whenever the items displayed exceed the panel dimension. However, when after this action, the contents of the panel is still partially hidden, a scrolling feature has been implemented coherently to all the panels in the FSDAS graphical interface.

- Due to the richness of data in the fishery domain, some item lists (e.g. species of a family) were extremely difficult to browse.

  We identified common item properties and created clusters of data attributes, which are initially collapsed. For instance the "Entity Information" panel has been provided with a _grid with grouping control_, which clusters the information according to the displayed properties (e.g. hasFamily, hasSpecies, etc). With this feature, the user can collapse each group as appropriate.

- In FSDAS v2, neither, the templates for querying the system, nor the results returned for a query, provided a faceting features, leaving the users with the burden to filter out the relevant information.

  In FSDAS v3 we identify properties of the queries and the result set, and use them to create facets for displaying element in tabs. The user is hence given a first quick view on what kind of items she/he is looking at.

6.3 Confidence in the validity of the returned result set

This evaluation aspect is defined as the support provided to end-users to assess the quality of the returned results. Annex 5 lists all proposals to improve FSDAS v3 compared to v2.

- Not all FSDAS users have the same familiarity with the provenance of the integrated data in the knowledge base. For them it is difficult to assess how relevant the results returned to their queries are.

  FSDAS v3 has planned to introduce a rating mechanism based on the feedback of the FSDAS user community. When a user is confident enough to express a value of quality, based on his/her experience, then this will be visible to support those who will retrieve the same result in the future.

6.4 Help and support

This evaluation aspect is related with the on-line or off-line support given to the users of FSDAS. ANNEX 6 lists all of the interventions needed to improve from FSDAS v2.

- FSDAS v2 did not offer on-line help, i.e. some "hints & tips" useful to the users when interacting with the web client.

  The GUI of FSDAS v3 has been enriched with a number of icons, positioned close to some of the panels in the interface, displaying a short explanation of the panel content when hovered with the cursor. Similar way of providing help, without leaving the main FSDAS interface, is inline suggestion for a valid input expected on text-fields .Still what a description of a functional component (e.g. Search Panel).

  Additionally, we added more and more elaborate warning messages in case user input was insufficient or other error conditions are met.

6.5 Terminology

This evaluation aspect is related with the labelling of the FSDAS graphical elements. ANNEX 7 lists all of the interventions needed to improve this aspect of FSDAS.

- In some GUI elements of FSDAS v2 misleading terminology was used. This was due to un-reflected usage of ontology terms in the user interface, some of which were confusing for users.

  In FSDAS v3 we improved this aspect by allowing terminology that better matches the requirements for FSDAS final users.

7 Conclusions

The work dedicated to the development of FSDAS v3 concerned one methodological aspect, one data integration aspect, and a number of technical aspects for software improvement.

The conclusions gathered after releasing FSDAS v2 stressed the overtime invested into aligning efforts from many different kinds of competencies aiming to understand the complexity of XML factsheet data structure. With FSDAS v3 we decided to improve the transition step of porting legacy data from FIES information systems to the FSDAS knowledge base. FIES experts worked jointly to design and implement web services as a uniform application interface to expose the data for the tripling process. Handling the complexity internally in the Fisheries division allowed allocation of more time to introduce FAO and NAFO catch records from statistical domain. The application knowledge base is hence enriched with more data cross-referencing each other, and thus enhancing better information retrieval experience for stock assessment.

The statistical dimension of fisheries data was missing from both FSDAS v1 and v2. It is indeed a key aspect in the scope of assessing fish stock depletion, and it has been integrated within the FSDAS v3 knowledge base. Initially, the study of the statistical catch records domain, led to the design of a new set of networked ontologies with other fisheries models from previous work. After this, FAO and NAFO catch records have been collected from the two organizations, and tripled according to the ontological models. Other statistical data has been collected from the domain of aquatic resource observations. This is a set of XML factsheets that has undergone the same procedure of domain conceptualization, data extraction and tripling Finally, the FSDAS knowledge base has been enriched with data integrated from more domains than the previous two releases. This will ensure more comprehensive data context when querying the system for fish stock depletion assessment.

When technically improving the FSDAS v3 prototype, our attention has been directed to fulfil end-users requirements gathered from feedback in the software evaluation. We worked on the flexibility of the layout of the graphical interface, as well as improving accessibility of the web client. Other efforts concerned the support provided to the users while they are performing their retrieval task with inline help features, improved context presentation when displaying the results of a user query etc. Also interaction with the web client has been improved, e.g. where the users were previously confused about the next step to take, we improved the workflow of the retrieval procedure.

Finally, the development of FSDAS v3 is a valuable effort to FIES division, because both the web services and the network of ontologies produced will serve other application development projects for Fisheries division, and provide benefit left by the NeOn project.

8 References

[AGMES]   FAO. Agricultural Metadata Element Set.
                 http://www.fao.org/aims/agmes_intro.jsp

[AIDA]   http://www.fao.org/fi/figis/devcon/schema/3_6/aida.xsd

[ASFIS]   L. Garibaldi, S. Busilacchi. ASFIS List of species for fishery
                 statistic purposes.
                 ftp://ftp.fao.org/docrep/fao/006/y7527t/y7527t00.pdf

[DC]   Dublin Core Metadata Initiative. http://dublincore.org/

[EDC]   Extended Dublin Core.
                 http://dublincore.org/schemas/xmls/qdc/2003/04/02/dc.xsd

[Fsdic]   FIGIS XML. List of elements.
                 http://www.fao.org/fi/figis/devcon/diXionary/figisdoc3.5.html

[FISTAT]   Fisheries and Aquaculture Department. Statistics.
                 http://www.fao.org/fi/website/FIRetrieveAction.do?dom=topic&fid=1
                 6062

[FSschema]   XML schema for Fisheries Fact Sheets.
                 http://www.fao.org/fi/figis/devcon/schema/3_6/fi.xsd

D7.1.2]   Revised specifications of user requirements for the Fisheries case
                 study.

[D7.2.3]   Initial Network of Fisheries Ontologies002E

[D7.2.4]   Initial Network of Fisheries Ontologies. Available from NeOn
                 project web site: http://www.neon-project.org (2009).

[D7.5.1]   Software architecture for the ontology-based Fisheries Stock
                 Depletion Assessment System (FSDAS). Available from NeOn
                 project web site: http://www.neon-project.org (2009).

[D7.6.1]   First prototype of the Fisheries Stock Depletion Assessment
                 System (FSDAS). Available from NeOn project web site:
                 http://www.neon-project.org (2009).

[D7.6.2]   Second prototype of the Fisheries Stock Depletion Assessment
                 System (FSDAS) Available from NeOn project web site:
                 http://www.neon-project.org (2009).

[D7.7.2]   Evaluation of the FSDAS second prototype and recommendations
                 to research. Available from NeOn project web site:
                 http://www.neon-project.org (2009).

9 Annex 1: How to install FSDAS v3

9.1 FSDAS Client Installation

No installation of any software is required. Users just need to direct their web browsers to the following URL:

> http://212.170.156.131:10000/FSDAS_web3/

9.2 FSDAS Server Installation

1. Download server installation files from:

http://212.170.156.131:10000/FSDAS_web3/FSDASINSTALLATION.rar

2. The Server Installation file contains the different files, which deploy the FSDAS prototype v2. In the following steps we describe how to deploy it in your server:

    a. Install the knowledge base files

        • Unzip the KB.zip

        • Remember the KB path for the OntoBroker server configuration (section 7.3)

    b. Unzip the file apache-tomcat-6.0.16.zip

    c. Deploy the Web application (FSDAS_web.war) within the webapps folder of the Apache Tomcat.

    d. Start Apache Tomcat.

9.3 FSDAS Server Configuration

In order to properly configure the FSDAS server we need to set some OntoBroker configurations on the server side to specify the location of the knowledge base or the license key.

1. Change the path of the knowledge base

    • open the OntoConfig.prp file (you can find it in configfiles.zip)

    • go to line 78 (H2.URL = jdbc:h2:file:kbase/edb)

    • replace "kbase" with the full path to your knowledge base folder, e.g. d:/stuff/kbase

    • I.e.: the line should in this case "H2.URL = jdbc:h2:file:d:/stuff/kbase/edb"

2. Change the location of configuration files

    • Put the configuration files (OntoConfig.prp and ob_prof.key.xml) in the same folder as your development application (Eclipse or Tomcat for example)

10 Annex 2: How to use FSDAS v3?

This Annex describes basic steps for using the second prototype of the Fisheries Stock Depletion Assessment System. This quick guide is intended for end-users in order to make easier their interaction with the system and making it possible to evaluate this second prototype in terms of functionality and usability.

This guide goes into the most relevant use cases that allow the user to experience the interaction with the FSDAS web-based client.

10.1 Launching the application

The second version of the FSDAS has been released as a Web application. To start the application you should use a web browser, such as Internet Explorer or Mozilla Firefox. The application has been tested with the latest versions of these Web browsers.

Currently the application is hosted in a server located in ATOS premises. In this guide we therefore provide the current URL, which will be replaced with one in FAO servers once the deployment of the application is done. The current URL is

http://212.170.156.131:10000/FSDAS_web3/

When the web page is loaded, the main desktop page is opened:

Figure 17: FSDAS web default perspective

The user interface is composed by the following widgets:

- Taxonomy Widget: It is located on the left hand side of the page. In this widget the taxonomy tree (i.e. species taxonomy, water area taxonomy...) is rendered. The data is extracted directly from FAO ontologies located on the server. The user is able select from a combo-box which taxonomy to visualize in the widget. When the user selects one node of the tree, the Entity Information Widget updates its content.

- Entity Information Widget: It is located on the left hand side of the page. This widget depicts the details of the item selected in the taxonomy widget. The details are extracted from the knowledge base, including object properties, attributes, annotations and its respective values.

- Search Widget: It is a textbox located at the top right of the page. The user may input free text in order to search ontological resources in the knowledge-base.

- Search Result Widget: It is the grid that shows the results after a search performed using the Search Widget. It is located in the right of the web page.

- Query Composition Widget: This is a panel that allows the user to dynamically compose queries to the knowledge base. The queries are formed dynamically (based on a configurable XML file located on the server). The possible queries are based on the ontology network stored in the data layer. When the user completes the required inputs and submits the query, the server side executes a query against the knowledge base using the facilities provided by the Data layer API.

- Query Results Widget: It is situated at the bottom centre of the page. This component uses a grid to display the query results returned from the query performed in the Query Composition Widget. Currently, the parameters shown in the grid are: Title of the resource, Dataset, index terms and the link to the resource. When the link is clicked, the resource is open in the assigned application (i.e. DOC files in Word, PDF in Acrobat Reader, etc.).

10.2 Execute Data Instance Query

ACTORS

- Fisheries scientist

DESCRIPTION

- User executes a query against the knowledge base (FAO ontology network). The query is built based on the ontologies (mainly application ontology) and the query choices are a fraction of the concepts and properties.

PRECONDITIONS

- All query values are selected

  o There is a data source selected within the taxonomy widget (Species, Water Area, Land Area...).

  o All fields belonging to the specific query are filled within Query Composition Widget.

TRIGGERING EVENT(s)

- Select the data source in the taxonomy widget.

Figure 18: FSDAS web Query Composition widget

- User clicks the "Send Query" button in the same line than the attributes filled in. The query sent to the server side collects the attributes selected by the Show column and with filled restriction.

POST CONDITIONS

- System displays the query results.

FLOW OF EVENTS

a. BASIC FLOW

- User selects the data source in the combo of the taxonomy widget (i.e. species).

- User goes to the Query Composition widget and selects the attributes he would query.

  o For instance, user selects to ask by "*hasLocalName*" attribute.

  o User set the input keyword for the local name (i.e. "*shark*", "whale", "foca", "pez", etc.).

- User click on the Validate Button.

- System displays results in the Query Results Widget.

- User can sort results in the Query Results Widget by clicking any of the columns title.

- User can click on any item showed in each result and access/open the result resource provided by the system in the associated application.
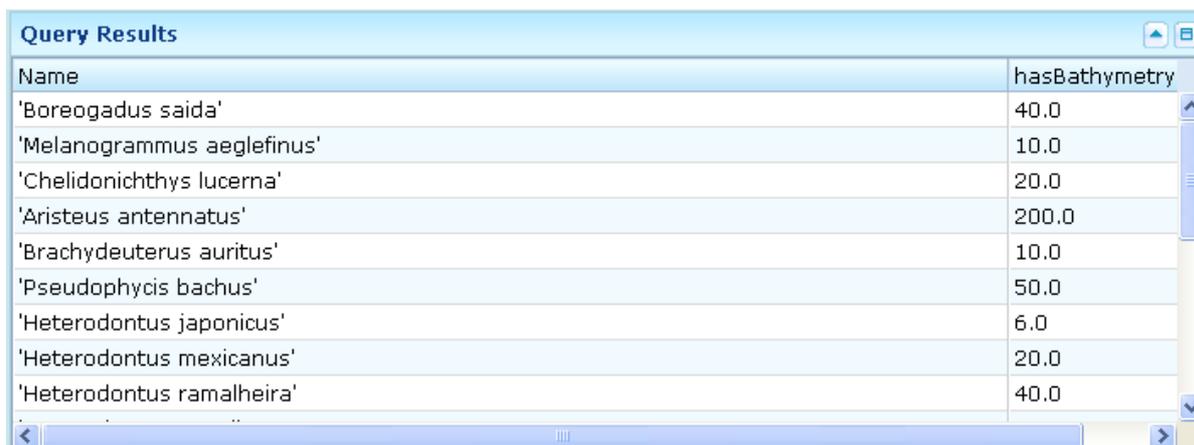
Figure 19: Query Results Widget

b. ALTERNATIVE FLOW: no results found

- System returns a no results alert.

RELATED USE CASES

- Any other Data Instance Query.

NOTES / ISSUES

For composed queries (various restrictions selected) it is compulsory to fill all fields of involved queries. The result will be the intersection of both results.

10.3 Execute Concept Search

ACTORS

- Fisheries scientist.

DESCRIPTION

- User executes Concept Search (Search for ontology content as concepts or data value).

PRECONDITIONS

- The ontologies are available at the server layer. These ontologies are loaded in Ontobroker and accessible via the API.

TRIGGERING EVENT(s)

Figure 20: FSDAS Keyword-Search entity widget

- User clicks the "Search Entity" button in the same line than the textbox is filled in.

POST CONDITIONS

- System displays the results of the search.

FLOW OF EVENTS

a. BASIC FLOW

- User goes to the Search Widget at the top-right of the page.

- User sets the input keyword for the search (i.e. oyster, 27.0, North-Atlantic etc.).

- User clicks on the "Search Entity" Button

- System displays results in the Search Results Widget.
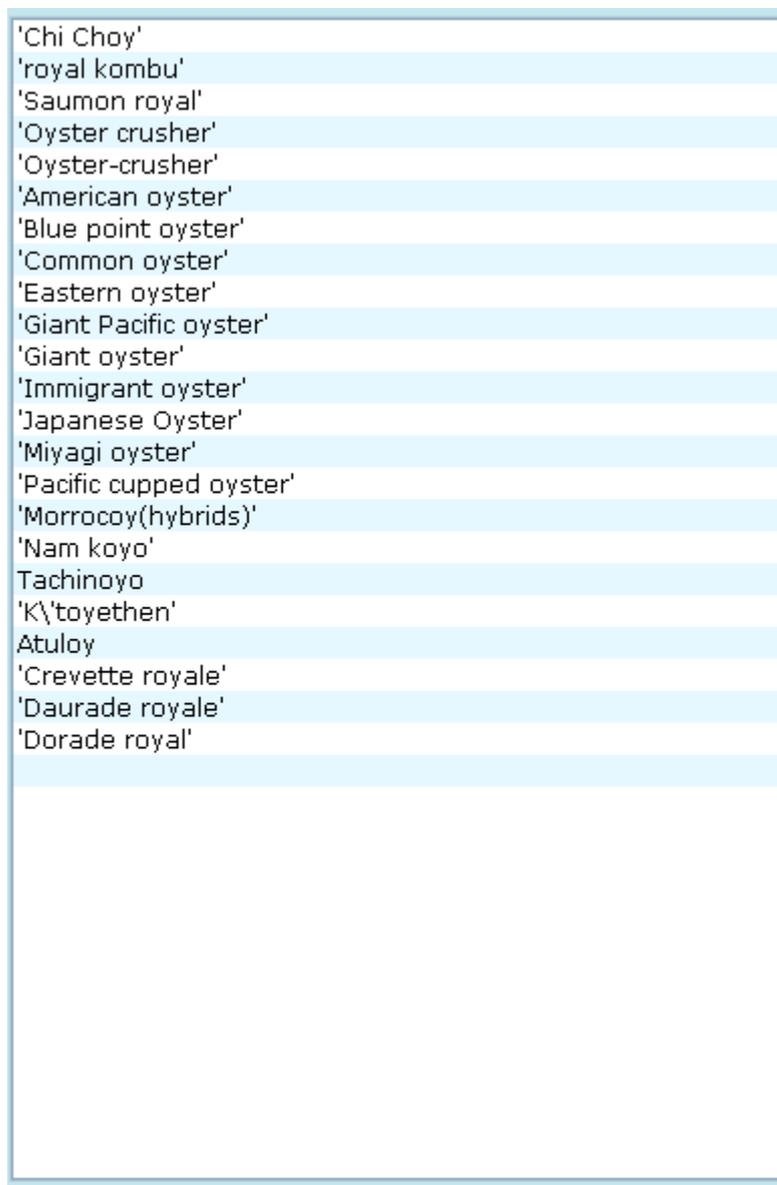
Figure 21: Keyword-Search result widget

b. ALTERNATIVE FLOW: no results found

- System returns a no results alert.

RELATED USE CASES

- Any other Concept / Data value Search.

NOTES / ISSUES

As a matter of example, the table below contains sample values for different concept search for testing purposes. It is especially advisable for users not familiar with FAO ontologies. The last column of the table contains possible values that can be used as input in the textbox:

Table 5: Example of test query values

Entity	Attributes	DataValue
Species Group	With identifier	8502.0
		8505.0
		8507.0
	With scientific name	MOLLUSCA
		CRUSTACEA
		MAMMALIA
Species Order	With identifier	8004.0
		8006.0
		8008.0
	With scientific name	LAMNIFORMES
		ORECTOLOBIFORMES
		CHARACIFORMES
Species Family	With identifier	7002.0
		7004.0
		7008.0
	With scientific name	NOTOCHEIRIDAE
		CYPRINODONTIDAE
		LEPISOSTEIDAE
Water Area	With Englishname	Pac antarctc
		Ind antarctc
		EC Atlantic
	With FAO Code	58.0
		88.0

10.4 Taxonomy Widget & Entity Information Widget

ACTORS

- Fisheries scientist

DESCRIPTION

- User visualizes some selected FAO taxonomies (Land Areas, Water Areas, Species), based on the ontologies provided by FAO. The widget includes paging functionality in the branches of the taxonomy tree.

PRECONDITIONS

- The ontologies are available in the server side.

- The ontologies are previously loaded in OntoBroker and extracted its associated instance taxonomy in the server business logic.

TRIGGERING EVENT(s)

- The user can select which taxonomy visualize using the combo box inserted in the Taxonomy widget.
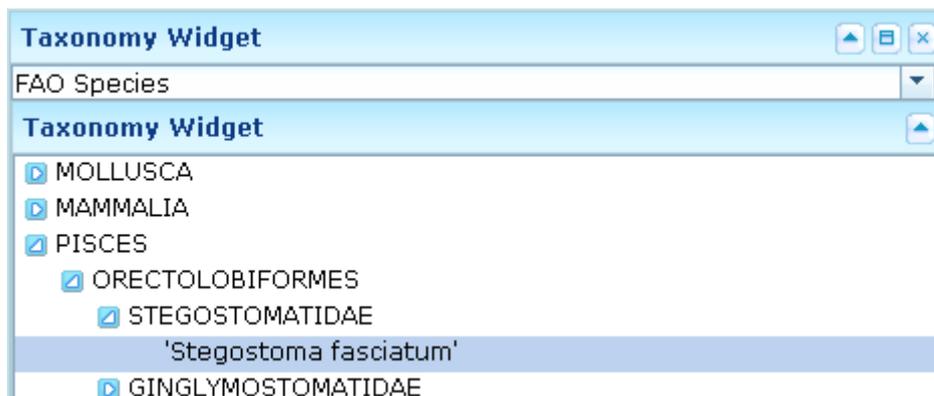
Figure 22: FSDAS web Instance Taxonomies Visualization

POST CONDITIONS

- System displays the instances taxonomy in the Taxonomy widget

FLOW OF EVENTS

a. BASIC FLOW

- User selects the ontology in the combo box.

- The ontology is visualized within the Taxonomy Widget.

- User clicks over the tree elements showed in the widget to expand or contract the different elements of the tree.

- The widget provides paging functionality when the taxonomy tree showed is bigger than 30 nodes.

- When the user double-clicks over any node of the taxonomy, the entity information widget update its content with the semantic information of the node extracted from the knowledge base.

Entity Information	
Attribute	Value
hasMeta	31005
hasNameScientific	'Stegostoma fasciatum'
hasNameFullEN	'Zebra shark'
hasCodeAlpha3	OSF
hasCodeTax	1070100101
hasNameFR	'Requin zèbre'
hasNameEN	'Zebra shark'
hasNameFullES	'Tiburón acebrado'
hasID	15438
hasNameES	'Tiburón acebrado'
hasNameFullFR	'Requin zèbre'
hasNameLongEN	'Zebra shark'
hasNameLongFR	'Requin zèbre'
hasNameLongES	'Tiburón acebrado'

Figure 23: Entity Information Widget

b. ALTERNATIVE FLOW: no results found

- The ontology is not available

- The entity information of the node is not found.

RELATED USE CASES

- Any other Ontology visualization.

- Any other Entity Information visualization as View ontological resource annotation.

11 Annex 3: Information Retrieval Process Bug Fix Board

Ticket	Issue	Proposed solution	Development Impact(H/M/L)	Priority
T1.1	When the result list is displayed, it is not intuitive that each list item is a link to the data provenance.	Make more explicit the possibility to click on a list item with an icon, or with the extended URL to the factsheet.	L	H
T1.2	The keyword based search does not allow search latin names	Make available the index of latin names for each entities of the knowledge base that have one.	M-H	H
T1.3	There is difference of result retrieved when search with the keyword "atun" and "atùn"	Make the index of names of the entities in the knowledge base invariant with respect to the charset used.	M-H	H
T1.4	The information listed in the "Entity information" panel are randomly shown every time an entity is selected		M	
T1.5	When the user search with keyword "seabass" or "bass" he/she has two different set of result	Improve the indexing algorithm	M-H	H
T1.6	User gets no result when he/she forgets to check the query	Do an automatic check when the input field is not empty.	M	M
T1.7	When selecting "Land Area" and "Water Area" taxonomy the "Query Widget" displays queries through properties which do no apply neither to Land or Water domain	Reproduce the same behavior for "Land Area" and "Water Area" that is shown when selecting either "FAO Specie" or "Aquatic Resource" taxonomy.	L	M
T1.8	When selecting "Land Area" and "Water Area" taxonomy any query triggered throws a message of exception	Fix this to a normal functioning condition that is shown when selecting either "FAO Specie" or "Aquatic Resource" taxonomy.	M	H
T1.9	User does not get any result for misspelling the input.	Support with either drop down list or auto-completion of the input text	M	M
T1.10	User does not get any result for the search shows case sensitive behaviour	Remove case sensitive from any search functionality based on input text.	L	M

12 Annex 4: Graphic Interface Bug Fix Board

Ticket	Issue	Proposed solution	Development Impact(H/M/L)	Priority
T2.1	"Enter" key does not trigger search request	Enable "Enter" key as alternative to clicking the interface button to trigger the query process.	L	M
T2.2	"Entity information" panel is too small to catch the attention of the user when seeking for selected entity information.	Have the "Entity information" panel self adapting to the screen size.	L	H
T2.3	"Entity information" panel is visually detached from the "Entity search" panel, and does not catch the attention of the user when seeking for selected entity information.	Move the "Entity information" panel in the middle of the screen.	L	H
T2.4	The window that opens to display the fact sheet from the selected result item is neither displaceable on the screen, nor resizable, nor is possible to reduce it for later usage.	Give the window displaying the fact sheet the behaviour of any other window in the system.	L	M
T2.5	The elements of the taxonomy only expand if the arrow on the left end of each item is clicked.	Make the taxonomy element expand also when the text is clicked.	M	M
T2.6	It is not clear for all users how to close the window displaying the fact sheet from the selected result item.	Provide the window displaying the fact sheet with a button to close it.	L	H
T2.7	The animation showing that the system is processing before providing a query result is not visible.	Display the animation displaying ongoing processing, at the center of the screen.	L	M
T2.8	It is not always clear that the "send query" button triggers the search as prepared in the "Query widget".	Enable "Enter" key or move the "send query" button in a place that follows more naturally the user action flow.	L	H

T2.9	The "Entity information" panel has pagination of the listed items, differently from all other panels that have a scrollbar when the number of items exceeds the one allowed by the panel size.	Provide the "Entity information" panel with scrollbars	L	H
T2.10	The "Taxonomy" panel pushes the ''Entity Information" panel out of the screen, when the user expands the tree nodes.	Provide the "Taxonomy" panel with vertical scroll bar	L	H
T2.11	The "Taxonomy" panel displays items names on two lines when they are too long to be contained in the panel size	Provide the "Taxonomy" panel with horizontal scroll bar	M	H
T2.12	When clicking on a root element in the "Taxonomy" panel, the information about included Order, Family and Species generates a long list of property with coded values difficult browse.	The item in the list should be clustered together.	M/H	M
T2.13	User feel lost of all the possible query he/she is presented at once.	Give more context to the interface so that the user is naturally driven through it. Use for example tabs, explanatory text, and simplified design (e.g. Google style)	M/H	M

13 Annex 5: Result Confidence Bug Fix Board

Ticket	Issue	Proposed solution	Difficulty (H/M/L)	Priority
T3.1	Some testers could not find what they expected to be part of the document result set.	Augment the data being part of the knowledge base.	H	L
T3.2	Some testers, with no specific background in aquatic resource, lacked a term of confidence to assess the validity of the retrieved document.	Introduce a form of rating/scoring the result so that is visible to the user the	H	L

14 Annex 6: Help and Support Bug Fix Board

Ticket	Issue	Proposed solution	Difficulty (H/M/L)	Priority
T4.1	When no checkbox is selected in the "Query Widget" the error message displayed does not give hints to solve the problem.	Provide a meaningful message to drive the user to check at least one item in the "Query Widget" panel.	M	M

15 Annex 7: Terminology Bug Fix Board

Ticket	Issue	Proposed solution	Difficulty (H/M/L)	Priority
T5.1	Trailing the property names with "has" is confusing for all the users not used to ontological terminology.	Use label that are more suitable for FI user classes	L/M	H