Project No. 249024

NETMAR

Open service network for marine environmental data

Instrument: *Please tick*	CA	✓ **STREP**	IP	NOE

ICT - Information and Communication Technologies Theme

D2.4.2 NETMAR system architecture – Authoritative specification

Reference: D2.4.2_System_Architecture_r1_20130215

Due date of deliverable (as in Annex 1): M0 + 36
Actual submission date: 15 February 2013

Start date of project: 1 February 2010 Duration: 3 years

Coastal and Marine Resources Centre (CMRC), University College Cork, National University of Ireland

Revision 1

Project co-funded by the European Commission within the Seventh Framework Programme (2007-2013)		
Dissemination Level		
PU	Public	X
PP	Restricted to other programme participants (including the Commission Services)	
RE	Restricted to a group specified by the consortium (including the Commission Services)	
CO	Confidential, only for members of the consortium (including the Commission Services)	

European Commission
Information Society and Media

	NETMAR Open service network for marine environmental data Project Reference: 249024 Contract Type: Collaborative Project Start/End Date: 01/03/2010 - 31/01/2013 Duration: 36 months
	Coordinator: Prof. Stein Sandven Nansen Environmental and Remote Sensing Center Thormøhlensgate 47, Bergen, Norway Tel.: +47 55 20 58 00 Fax. +47 55 20 58 01 E-mail: stein.sandven@nersc.no

Acknowledgements

The work described in this report has been partially funded by the European Commission under the Seventh Framework Programme, Theme ICT 2009.6.4 ICT for environmental services and climate change adaptation.

Consortium

The NETMAR Consortium is comprised of:

- Nansen Environmental and Remote Sensing Center (NERSC), Norway (coordinator).
  Project Coordinator: Prof. Stein Sandven (stein.sandven@nersc.no)
  Deputy Coordinator: Dr. Torill Hamre (torill.hamre@nersc.no)
  Quality Control Manager: Mr. Lasse H. Pettersson (lasse.pettersson@nersc.no)
- British Oceanographic Data Centre (BODC), National Environment Research Council, United Kingdom
  Contact: Dr. Roy Lowry (rkl@bodc.ac.uk)
- Centre de documentation de recherche et d'expérimentations sur les pollutions accidentelles des eaux (Cedre), France.
  Contact: Mr. François Parthiot (Francois.Parthiot@cedre.fr)
- Coastal and Marine Resources Centre (CMRC), University College Cork, National University of Ireland, Cork, Ireland.
  Contact: Mr. Declan Dunne (d.dunne@ucc.ie)
- Plymouth Marine Laboratory (PML), United Kingdom.
  Contact: Mr. Steve Groom (sbg@pml.ac.uk)
- Institut français de recherche pour l'exploitation de la mer (Ifremer), France.
  Contact: Mr. Mickael Treguer (mickael.treguer@ifremer.fr)
- Norwegian Meteorological Institute (METNO), Norway.
  Contact: Mr. Øystein Torget (oysteint@met.no)

Author(s)

- Anthony Patterson, CMRC, a.patterson@ucc.ie
- Torill Hamre, NERSC, torill.hamre@nersc.no
- Declan Dunne, CMRC, d.dunne@ucc.ie
- Peter Walker, PML, petwa@pml.ac.uk
- Yassine Lassoued, CMRC, y.lassoued@ucc.ie
- Jorge de Jesus, PML, jmdj@pml.ac.uk

Document approval

- Document status: Revision 1
- WP leader approval: 2013-02-15
- Quality Manager approval: 2013-02-15
- Coordinator approval: 2013-02-15

Revision History

Issue	Date	Change records	Author(s)
Draft 1	2012-12-20	Draft based on final version (2.2.2)	A. Patterson
Draft 2	2012-12-20	Validation of front-end, views	A. Patterson
Draft 3	2013-01-22	Introduction, conclusions, summary	A. Patterson
Draft 4	2013-01-24	PML input	P. Walker
Draft 5	2013-01-29	NERSC input, final version for external review	T. Hamre A.Patterson
1	2013-02-15	Incorporated external reviewer comments	A. Patterson D. Dunne P. Walker
1	2013-02-15	Final version approved by the coordinator.	T. Hamre

Executive Summary

NETMAR has developed a pilot European Marine Information System (EUMIS) for searching, downloading and integrating satellite, in situ and model data from ocean and coastal areas. This report addresses the NETMAR objective of designing the system architecture using service oriented architecture (SOA) principles and patterns while fulfilling user requirements from documented use cases in four pilot applications:

- Arctic Sea Ice and Met-ocean Observing System
- Oil spill drift forecast and Shoreline Cleanup assessment services in France
- Ocean Colour – Marine Ecosystem, Research and Monitoring
- The International Coastal Atlas Network (ICAN)

We begin by introducing the core capabilities which NETMAR has built upon through research into the state of the art, interaction with stakeholders through discussion and testing, and the application of service oriented architecture principles. These represent specific areas of functionality in which the NETMAR partners have existing expertise. By bringing together expertise in the areas of portals, semantics, orchestration and geospatial standards through the medium of a service oriented architecture, NETMAR's aim is to show how its user groups can significantly improve their performance in the tasks represented by the four pilot applications.

We expand on the core capabilities of NETMAR by illustrating the concrete architectural decisions which have been made to date. We look at these from a number of complementary viewpoints. These viewpoints are based on the Orchestra adaptation of the RM-ODP reference architecture. We look at the data, metadata and data flow of the system from the Information viewpoint. We examine some of the detailed infrastructure and infrastructure support decisions from the Engineering viewpoint, and we list some of the main technology and standardisation decisions made from the Technology viewpoint.

The Enterprise viewpoint is represented by an analysis of the key drivers of the NETMAR architecture. These are presented from the perspective of three principle stakeholder groups. The research community is interested in NETMAR's outcomes, as they will form a basis for validating and extending good practice in the design of environmental information systems. The service and data provider community will want to know how to build upon NETMAR's capabilities in order to add new value. Finally, NETMAR's end users from the four pilots will want to see a set of tools which offer a significant improvement on how they perform their work. Within the Enterprise View, we also examine the requirements derived from the NETMAR use cases, with a view to relating these to specific decisions within the NETMAR architecture, and we look at some of the previous projects from which NETMAR has drawn inspiration.

The Service viewpoint shows how our design reflects service oriented architecture principles. It does this by referencing existing SOA design patterns which represent documented good practices in SOA design derived from experience gathered from the implementation of many different SOAs. We show examples of patterns in use within NETMAR and illustrate how we follow the pattern, how it helps achieve our desired results, and what adaptations, if any, we needed to make to the basic pattern.

The as-built architecture at the time of writing of this report does not completely conform to the viewpoints described. Where there is work to be done in building out the architecture completely, we have attempted to highlight this and make specific recommendations about how the individual components could be integrated to provide a more coherent system.

Contents

1 Introduction

1.1 Background

NETMAR has developed a pilot European Marine Information System (EUMIS) for searching, downloading and integrating satellite, in situ and model data from ocean and coastal areas. It is user-configurable system offering service discovery, access and chaining facilities using OGC, OPeNDAP and W3C standards. It uses a semantic framework coupled with ontologies for identifying and accessing distributed data, such as near real-time, model forecast and historical data. EUMIS also enables further processing of such data to generate composite products and statistics suitable for decision-making in diverse marine application domains.

The system architecture has been strongly guided by the use case definitions work package (WP1) which has defined user requirements. Section 4.4 illustrates how user requirements have influenced design decisions. Strong interactions have also been made with the ontologies (WP3), semantic framework (WP4), service networks (WP5) and EUMIS development (WP6) work packages in order to realise an open, robust and scalable system architecture where the various sub-systems and components integrate.

This report is based on the Final Version of the Architecture Specification [PHD+12]. While the Final Version provided a baseline of decisions made on how to integrate the NETMAR components into a coherent system, the Authoritative Specification is intended to measure the effectiveness of those decisions. The intent is to take a set of guidelines grounded in theory which have been applied to a particular problem domain, and measure how well they worked in practice. The authoritativeness of the document is in proportion to how well it demonstrates that the practical application of the design decisions demonstrated their effectiveness.

1.2 Objective of this report

The primary objective of this report is to provide a sound basis for the production of good practice guidelines for the implementers of an environmental information system. By documenting our decisions from a number of complementary viewpoints, explaining the motivations and rationale for these decisions, and relating them to existing good practice in software engineering, we hope to make clear why we have chosen certain design paths over others. By documenting how those decisions were validated by meeting specific customer needs, we give future designers of such systems practical guidelines on how and where to follow our example.

1.3 Terminology

Capability – the ability to perform actions based on expertise and capacity. A **Service** exposes a **Capability**

Computational viewpoint – is one of the viewpoints of RM-ODP, which specifies the system as a set of objects (or services) interacting through interfaces. This view describes the functionality of the system and the functional decomposition.

Engineering viewpoint – is one of the RM-ODP viewpoints, which focuses on the mechanisms and functions required to support distributed interaction among objects in the system.

Enterprise – An undertaking, especially one of some scope, complication, and risk.

Enterprise architecture – A rigorous description of the structure of an enterprise.

Enterprise viewpoint – is one of the viewpoints of RM-ODP, which has focus on the overall purpose, scope and policies for the system, and describes its business requirements and how these can be met.

Information viewpoint – is one of the viewpoints of RM-ODP, which defines the semantics of the information that the system will handle and the needed information processing. This viewpoint describes the information handled by the system as well as the structure and type of supporting data.

Pattern - a general reusable solution to a commonly occurring problem within a given context in software design.

Pattern Language - a structured method of describing good design practices within a field of expertise.

Pilot - The implementation of one or more use cases for a set of users, all within the framework of EUMIS and using the semantic resources that are developed in the project. It is a proof of concept for EUMIS as a foundation for development of marine environmental information systems.

Reference Model – an abstract framework for understanding the elements of the system and the relationships between them.

Requirement – a documented description of what a system must do or how it must do it.

RM-ODP – Reference Model for Open Distributed Computing is a reference model for developing and documenting the architecture of distributed information systems [ISO10]. The RM-ODP defines the system architecture through five complementary but consistent viewpoints: the enterprise viewpoint, the information viewpoint, computational viewpoint, the engineering viewpoint, and the technology viewpoint.

Scenario - An instance of a use case. A scenario represents a concrete sequence of interactions between one or more actors and the system. [BD10]

Service - A mechanism to enable access to one or more capabilities, where the access is provided using a prescribed interface and is exercised consistent with constraints and policies as specified by the service description [OAS06].

Technology viewpoint – is one of the RM-ODP viewpoints, which describes the technologies chosen to implement the system.

UML – Unified Modelling Language is standard modelling language for modelling different aspects of computer system developed and maintained by the Object Management Group [UML23]. UML offers an expressive diagram notation that can be used to depict structure, behaviour and interaction of the various components of the system in rich detail, as well as a language (the Object Constraint Language, OCL) that can be used to express further conditions and constraints that must be fulfilled.

Use case - A general sequence of interactions between one or more actors and the system. See also scenario. [BD10]

Validation - checking that a software system meets specifications and fulfils its intended purpose.

2 Core Capabilities

This section describes the core areas of functionality which NETMAR will build upon. These capabilities will be essential to meeting the core system requirements identified through interaction with users in four pilot applications [PTS+11]:

- Arctic Sea Ice and Met-ocean Observing System
- Oil spill drift forecast and Shoreline Cleanup assessment services in France
- Ocean Colour – Marine Ecosystem, Research and Monitoring
- The International Coastal Atlas Network (ICAN)

The three main areas of functionality in EUMIS are:

1. **Common user interface.**
   This is the users' entry point to EUMIS, providing access to all products and services through a web browser interface.

2. **Semantic Web Services and Ontologies.**
   This utilises semantic reasoning for smart discovery of relevant data products and services. It is also used to support semantic validation of service chains in the service chaining editor.

3. **Service chaining editor and workflow engine.**
   This includes a service chain editor, semantic validation of service chains, the execution of chained services in a workflow orchestration engine, and the management of uncertainty propagation through the service chain.

These three main areas of functionality are described in the following sections. The common user interface is first outlined as it is the users' entry point to the EUMIS.

2.1 Common User Interface

The EUMIS portal offers a common user interface to all products and services that are offered by the providers through the service network. This user interface must also provide information about which products and services are available, how to access and use them, how to contact providers to obtain additional information or request new or customised products or services, and how to register as a user for getting access to restricted facilities.

2.1.1 *Portal User Interface*

The EUMIS portal must offer all of its products and services through a common user interface. The user interface (UI) must satisfy the following requirements:

- It must be consistent, i.e. have a common look and feel throughout all pages of the portal.
- It must be easy to navigate and find relevant information, whether it is about specific products or services, or on the technologies that have been used to set up EUMIS. New users must be able to quickly find what they are looking for.
- It must be easy to operate the pilots, also for new users.

2.1.1.1 Use cases

The first operation a new user will perform is to look up information about the products and services offered by EUMIS. The user must be able to look up this information by topic (e.g.

application domain) or by searching using free text. When navigating by topic and subtopics, the users must always be able to return to the previous or top level with one mouse click.
Once a user has identified which pilot is of interest, he will access it and start looking at products and services being offered. He can select which types of products he wants, and EUMIS will display it in a GIS map viewer. He can also choose to run a processing service and when the result is ready, EUMIS will notify the user, and the computed parameter can be displayed in the GIS viewer together with the previously retrieved products.

2.1.1.2 Components

The Wiki will hold all information about products, services and technologies for the pilots (Figure 2-1). It will support easy navigation through the wiki pages, which are organised in a hierarchy (Figure 2-2), as well as free text search. The service developers can update their respective pilot pages, and users can be permitted to post comments to the content. The Wiki is publicly available.

Pilots are comprised of a GIS map viewer component, and optionally a search and discovery client, a service chaining editor and a workflow engine. Together these components provide the functionally requested by the targeted users in the application domain (e.g. Arctic sea ice monitoring and forecasting).

2.1.1.3 Rationale

The rationale for choosing a wiki as the mechanism for user support, is that this approach is familiar to most users from other systems. The wiki can be structured as a set of related web pages that can be interlinked, and they are searchable by the user. The current position in the wiki is shown in a tree-structured menu or a "breadcrumb" trail shown at the top of all pages. This allows the user quickly to find relevant information.

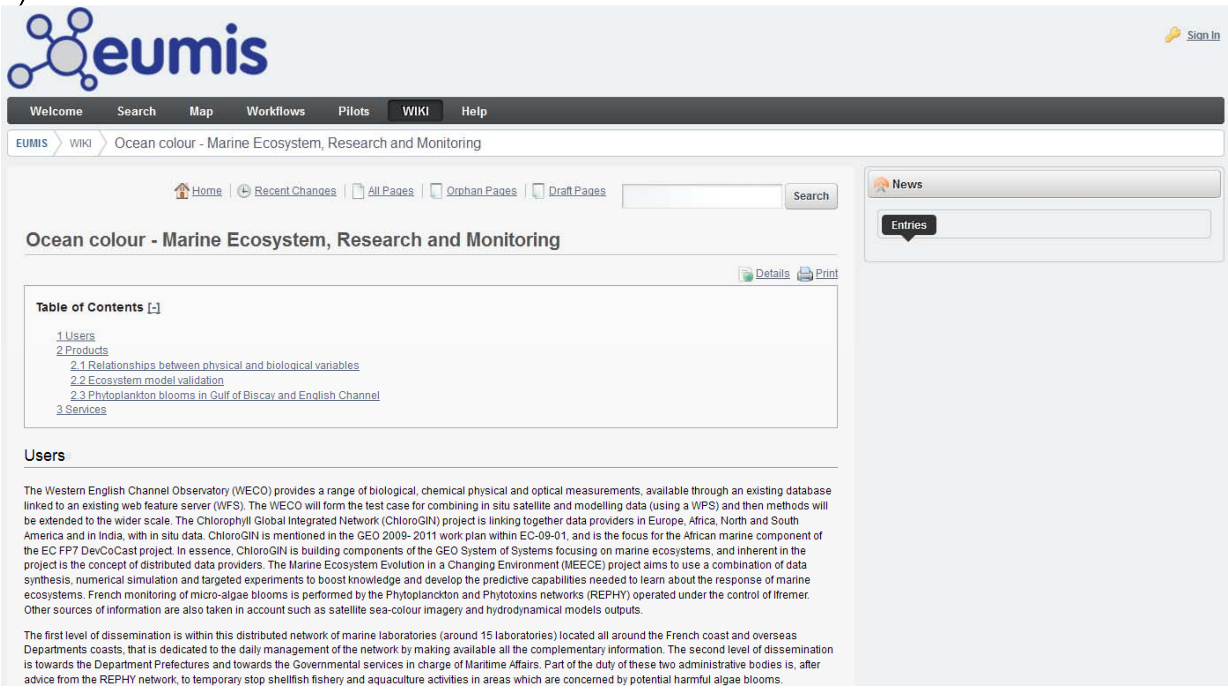
The criteria for choosing a portal framework have been:

- It must support the leading portlet standards, notably JSR-168 and JSR-286. This will ensure a high degree of flexibility in developing portlets, e.g. that a number of programming languages can be used to develop EUMIS components.
- It must be an open source framework. This to ensure that new service providers can set up their own portal based on it, and that the developed portal and components can be sustained.
- It must support a role-permission based user model, where users can be grouped in communities and given permissions to access products and (possibly) execute service chains based on their membership in a community. If needed, it must also be possible to refine the permissions for a user individually, if specific users within a community needs more fine grained permissions to access products and services.

Based on this, we have chosen the Liferay Community Edition version 6, which is an open source framework which is being actively developed. Liferay has a large user and developer base, offers a number of portlets built into the distribution (including e.g. wiki and forum), Liferay also has a community based user model, where permissions can both be assigned based on membership in a community, but also be refined for individual users, if needed.

Figure 2-1 The Wiki home page shows the top level wiki pages; one page for each pilot and supportive wiki pages for EUMIS components, metadata and technologies & tools. Clicking on the heading for pilot 3 (marked with arrow) will take the user to the wiki page for this pilot.

a)



b)

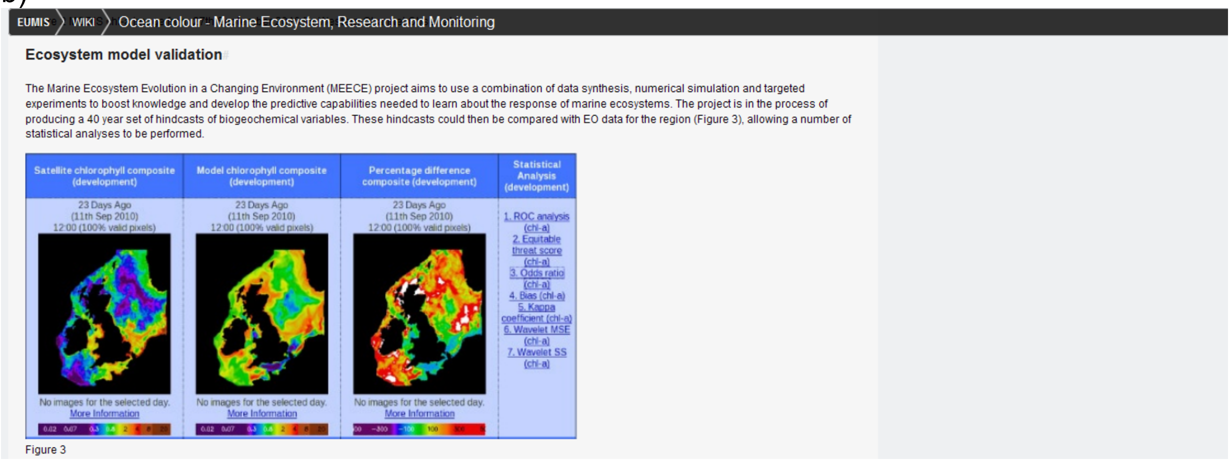


Figure 2-2 The wiki page for pilot 3, Ocean colour – Marine Ecosystem, Research and Monitoring has a Table of Contents at the top allowing user to quickly jump to topics of interest. By clicking on the heading for Ecosystem modelling (marked with arrow in (a)), the browser will scroll to this topic (b). Note that current location in the wiki is always shown at the top of the screen so the user can quickly go back to earlier topics or the wiki home page.

2.2 Semantic Web Services and Ontologies

The NETMAR system has a requirement for semantic resources to support discovery and use of both data and services. The Global Spatial Data Infrastructure cookbook describes three levels of metadata, namely: discovery, exploration, and exploitation metadata [GSDI09]. A draft INSPIRE report on metadata also names these three levels of metadata, in this case: discovery, evaluation, and use [INS07a]. For the purposes of the NETMAR project, two levels of metadata were used and named: discovery and usage metadata. NETMAR discovery metadata chiefly concerns discovery metadata with some additional exploration/evaluation metadata elements. NETMAR usage metadata chiefly concerns exploitation/use metadata and was researched as a distinct topic because of the special requirement to support semantic verification of service chains, ensuring that inputs and outputs are compatible.

2.2.1 *Data and service discovery*

Data and service discovery in the EUMIS will be semantically-enabled through the use of multi-lingual and multi-domain ontologies, and the semantic framework. The objective is to allow users to search resources by meaning rather than by mere keywords. This would, in particular, solve the problem of searching across different languages and/or domains. Data discovery will also support distributed catalogue services, thus allowing different systems, e.g. coastal web atlases, to be connected and uniformly accessed. Specific requirements are:

- Support for smart search functionality, i.e. ability to search a keyword based on semantics.
- Ability for smart search queries by parameter name, allowing datasets with related parameters to be found.
- Ability to support queries by parameter name combined with geographic area and/or time range.
- Discovery-level metadata to support data and service discovery.
- Usage-level metadata sufficient to support service chaining validation.
- Support for multi-lingual and multi-domain ontologies.
- Support for ontology browsing.
- Support for metadata visualisation.

2.2.1.1 Discovery Metadata

The overall discovery metadata model is based on INSPIRE compliant metadata records. However, to better support user discovery of datasets and services, topics are organised into different categories, depending on their purpose (e.g. themes, instruments, parameters, etc.). We call these topic categories “facets” as they provide different perspectives for searching data. The commonly agreed facets are the ISO 19115 and the SeaDataNet keyword types. More specifically, in the NETMAR project, the facets required by the pilots are:

- Theme/Discipline (parameter)
- Instrument
- Platform/Platform Class
- Project
- Vertical coverage
- Location

Two of the required facets, theme and discipline, may be considered to be parameter descriptions at different levels of granularity. However, for discovery metadata the conventional thinking within the oceanographic data management community has been that coarse granularity is required for the discovery-level parameter facet (i.e. parameter disciplines) [LLC+11].

2.2.1.2 Discovery use cases

Data and service discovery will be enabled mainly through two use cases: ontology browsing and smart search.

2.2.1.2.1 Ontology browsing

Ontology browsing is the process of exploring and searching the ontologies. It allows users to graphically view the available topics, their meanings and how they relate to each other, and to identify their topics of interest (e.g. Figure 2-3 and Figure 2-4). With ontology browsing, topics are organised using facets. The purpose of ontology browsing is to find in an accurate way datasets or services of interest.

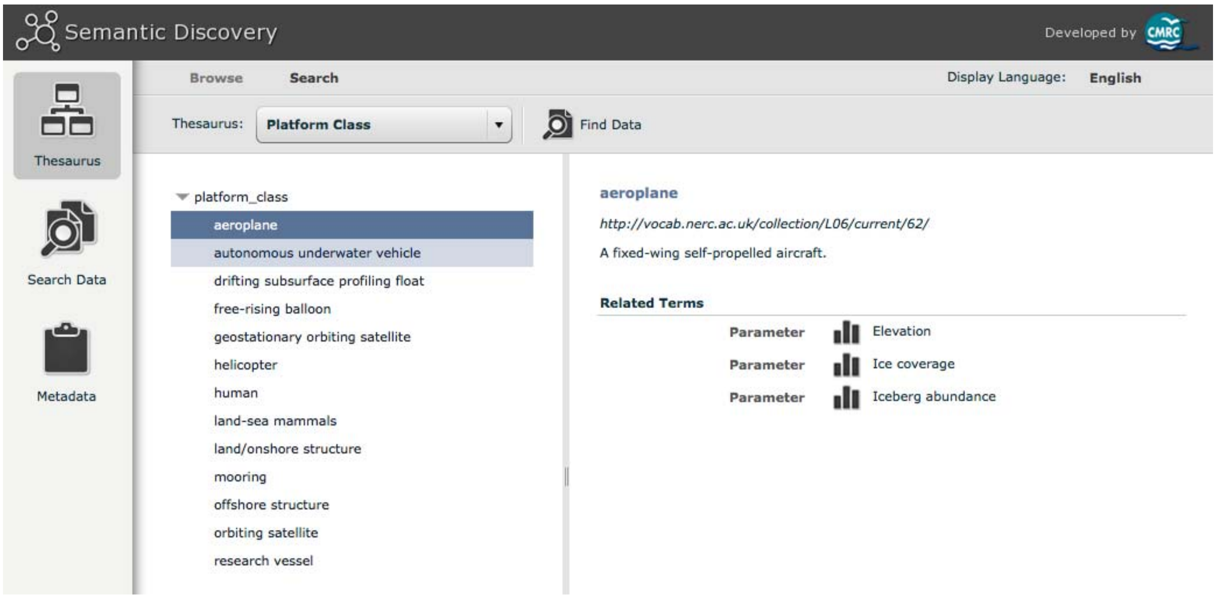


Figure 2-3 Example ontology browser interface

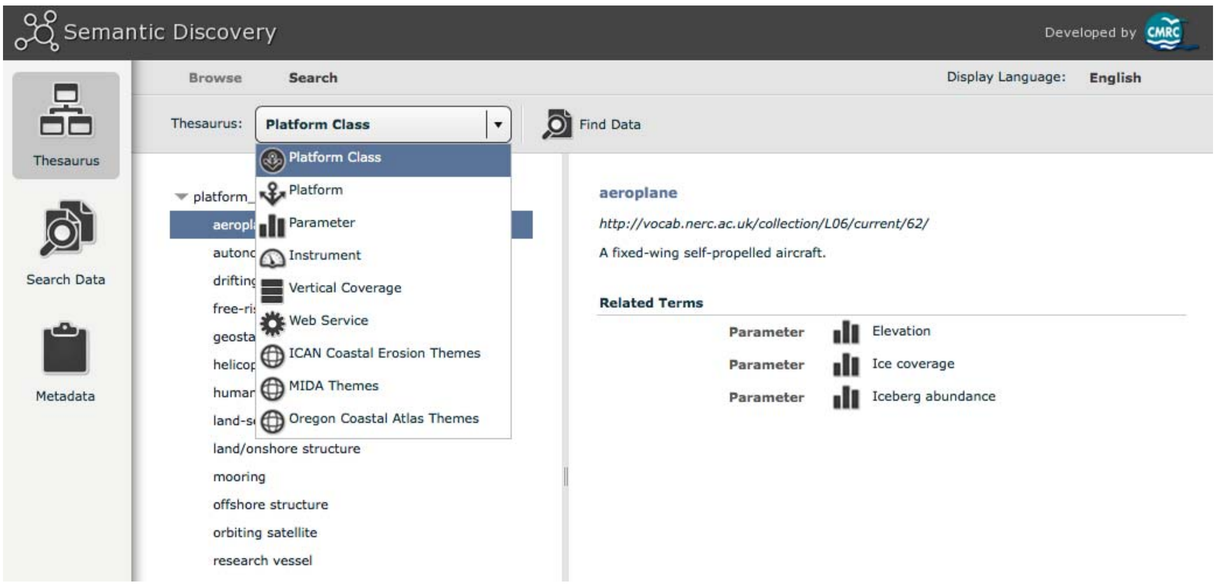


Figure 2-4 Example facet search options in an ontology browser

2.2.1.2.2 Smart search

Smart search is the ability of a discovery engine to locate target documents that do not include either an exact or fuzzy match to a user-supplied text string or user-supplied keyword. This type of search is based on the “meaning of a keyword” rather than on textual matches. The idea is to exploit the available semantic relationships between ontology terms in order to make data search more accurate and complete. For instance, if you search “elevation”, you would also get “bathymetry” and “depth contours” as these are narrower terms than “elevation.” Or if you search “seafloor”, you would also get datasets tagged as “seabed”, as these terms are synonyms.

Smart search can also be applied to support multi-lingual data discovery, e.g. the English term “water” will map to the French term “eau”, to the Spanish term “agua”,etc. Multilingual support is also important for system (e.g. coastal atlas) interoperability, and for support of pollution monitoring and emergency management in the coastal areas of Europe.

Smart searches may be organised through the use of controlled vocabularies, in which case the user is provided with a collection of drop-down lists from which to choose. However, they can also be totally open with the user supplying free text into a simple box such as the well-known Google search engine interface.

Smart discovery should support faceted search. There are a number of use cases for faceted smart search based on keywords taken from controlled vocabularies [LLC+11]:

- **Simple search**: In this case both the search interface drop-down list and the target documents are populated using keywords from a single controlled vocabulary for each facet. The user selects one or more keywords from the controlled vocabulary which are then matched against exact matches in the metadata.

- **Thesaurus search**: In this case the search interface drop-down list for a facet is initially populated by top concepts from a thesaurus. The search may be based directly on these or the user may be given the option to narrow their search by drilling down into the thesaurus. Metadata document markup is based on bottom concepts from the thesaurus. The semantic engine underpinning the search translates the user’s sections into a list of bottom concepts that are then matched against the metadata documents.

- **Orthogonal search**: In this case the search interface selection for a facet is again populated from either a simple controlled vocabulary or a thesaurus. However, in this case the metadata document contains no markup for that facet. The semantic engine initially translates the user selection into a set of bottom concepts within that facet. These are then translated through a mapping into a list of concepts for a facet that is included in the metadata markup. For example, a user may select ‘thermosalinograph’ from an instrument discovery facet. Mappings to a parameter facet would translate this to ‘temperature’ and ‘salinity’ and mappings to a vertical spatial coverage facet would translate it into ‘sea surface’. The semantic web service would therefore return metadata records marked up with ‘temperature’ or ‘salinity’ and ‘sea surface’ vertical coverage.

- **Spatial search**: In this case the drop-down list is populated by terms from a gazetteer that have spatial coverage geometry. The metadata documents are populated with spatial coverage geometry co-ordinates. The search client translates the user-selected name into a geometry which is supplied to a spatially aware search of the metadata markup. For example, the gazetteer term geometry may be translated into bounding box co-ordinates that are fed into search.

2.2.1.3 Components

The smart search and discovery client includes two main features (Figure 2-5): an ontology browser and a smart search interface (discovery GUI). It also supports a metadata viewer feature, enabling subsequent visualisation and/or download of discovered datasets or services.

The client relies on two services: a semantic web service (SWS) and a CSW[1] mediator (CSWM), as illustrated in Figure 2-5. The semantic web service (SWS) provides a high-level interface for retrieving knowledge from the NETMAR ontologies. The operations supported by the SWS are high-level and easy-to-use operations oriented towards querying SKOS thesauri. They constitute the common operations required by most semantics-based components such as the CSW mediator, the ontology browser, and the discover GUI.

In the back-end, ISO 19115/19119 metadata encoded in ISO 19139 are delivered by distributed catalogues services. CSW version 2.0.1, CSW version 2.0.2, and CSWM will be supported.

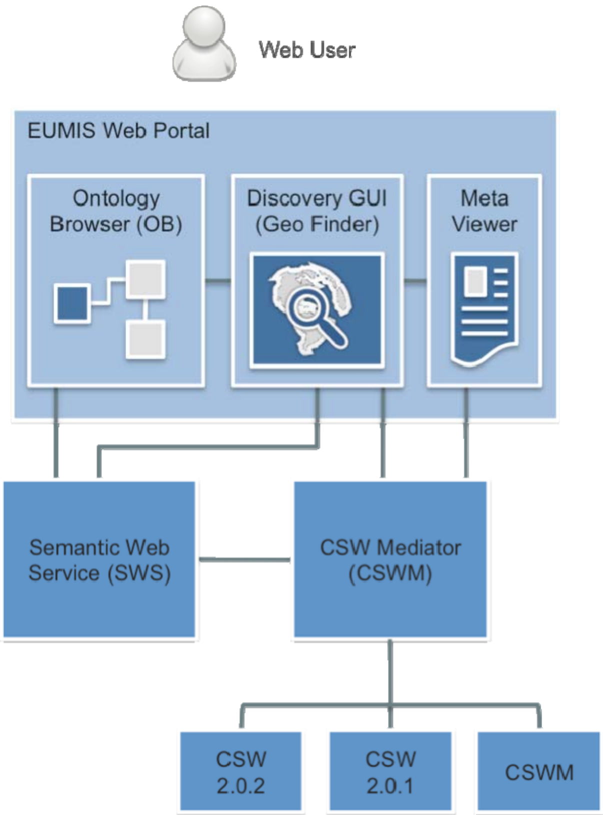


Figure 2-5 Discovery components in the EUMIS

2.2.1.4 Rationale

Metadata uses ISO standards, which ensures interoperability of metadata. ISO 19115 is used to describe datasets and/or dataset collections, while ISO 19119 is used to describe service metadata. This metadata is encoded in ISO 19139 XML. All these standards are also mandated by INSPIRE.

OGC CSW enables interoperability of metadata catalogues over the web. This distributed metadata catalogue approach ensures metadata can be maintained locally within an organisation, while still ensuring open publication of this metadata catalogue outside that organisation. CSW is also mandated by INSPIRE.

1 Catalogue Service for the Web (OGC specification)

CSW Mediator (CSWM) is a layer above CSW which enables smart searching of CSW metadata catalogues. As the CSW standard does not directly support smart search, we need this extra layer above CSW to support this functionality.

The semantic web service (SWS) provides a high-level interface for retrieving knowledge from the NETMAR ontologies. In fact, various components of EUMIS require semantic capabilities, e.g. service chaining, ontology browsing, smart search, etc. The types of semantic operations required are common to all these components (e.g. Get Concepts, Get Related Concepts, Get Concept Schemes, etc.) and require reasoning over the ontologies. Because, these NETMAR components require common semantic operations, it was decided to pool these operations into a discreet and reusable web service. This web service can be exploited by other components or tools in the future which require knowledge from ontologies.

The smart search, discovery clients, and metadata viewer are provided to support the NETMAR discovery use case, where the objective is to allow users to search resources by meaning rather than by mere keywords.

2.2.2 *Data and service usage*

2.2.2.1 Usage metadata

Usage or use metadata concerns specific metadata about the actual data format or service itself, e.g. fine-grained parameter names. Its overall model is based on data specification level. In essence, it contains information about how to "use" the data or service. Within NETMAR, usage metadata is required to support semantic verification of service chains, ensuring that inputs and outputs are compatible. More specifically, the usage metadata required by the pilots are:

- parameter
- unit of measure

Processing services should make use of usage metadata where available to ensure that if they are passed a parameter with a different meaning, for instance sea surface temperature (SST) instead of chlorophyll, they raise a suitable exception rather than returning invalid results. Also, for example, if a processing service calculates sea water density it must be able to verify that the input parameters supplied are pressure in decibars, practical salinity in dimensionless units and water body temperature in degrees Celsius or risk generating a meaningless result. Consequently, a mechanism is required for semantic labelling of service inputs and outputs with the parameter measured and its unit of measure [LLC+11]. Use cases for usage metadata in the context of service chaining editor and workflow engine are described in Section 2.3.

2.3 Service chaining editor and orchestration engine

To support service chaining in EUMIS, we need two main parts of functionality:

1. The ability to define a new service chain and store it for later use.
2. The ability to execute an existing service chain and visualise the result.

The first part will be realised by a service chaining editor, while the second part will be realised by an orchestration engine (a.k.a. workflow engine). These components are described further below.

While the service chaining editor is designed to make it easier for non-technical users to construct service chains, it is recognised that this will not be a trivial task, even with the help of the editor. Therefore NETMAR is providing tutorials on the EUMIS portal with example workflows and a short video showing the editor in use.[DJW11]. The results of workflows generated by the service chaining editor can be downloaded for local use as well as passed to a web based visualisation service.

2.3.1 *Service chaining editor and orchestration engine*

The service chaining editor must satisfy the following requirements:

- It should be accessible through all commonly used modern web browsers (e.g., IE8, Firefox 10 and above, Safari, etc.).
- The interface should be simple so that a non-technical user can build workflows.
- The interface should provide save and load functions.
- The editor should make use of semantic information (i.e. usage metadata) where available to guide and constrain the user in making correct workflows.

The orchestration engine must satisfy the following requirements:

- It must function independently of the other portal components.
- It must be accessible via a web interface.
- Workflows published to the orchestration server should be accessible as if they were OGC WPS processes.
- Processing services should make use of semantic metadata (i.e. usage metadata) where available to ensure that if they are passed a parameter with a different meaning, for instance sea surface temperature (SST) instead of chlorophyll, they raise a suitable exception rather than returning invalid results.
- Processing services should be able to understand and pass on uncertainty data through the chain.

2.3.1.1 Service chaining use cases

2.3.1.1.1 Build service chain

The Graphical User Interface (GUI) shown in Figure 2-6 illustrates that the editor must have a simple interface where services are selected from drop-down menu, and services are connected (i.e. chained) by drawing a line from the output field of the first to the input field of the next service. The Load and Save functionality must also be readily available e.g. as buttons, on the main screen. If a service has other input, such as a smoothing factor in the case of the r.fillnulls service, the user must also be able to set this by clicking at the parameter name inside the service symbol. (r.fillnulls is a service wrapper around the GRASS GIS function of the same name which fills no-data areas in raster maps using spline interpolation.) Figure 2-7 shows how the resulting service chain may be exported.

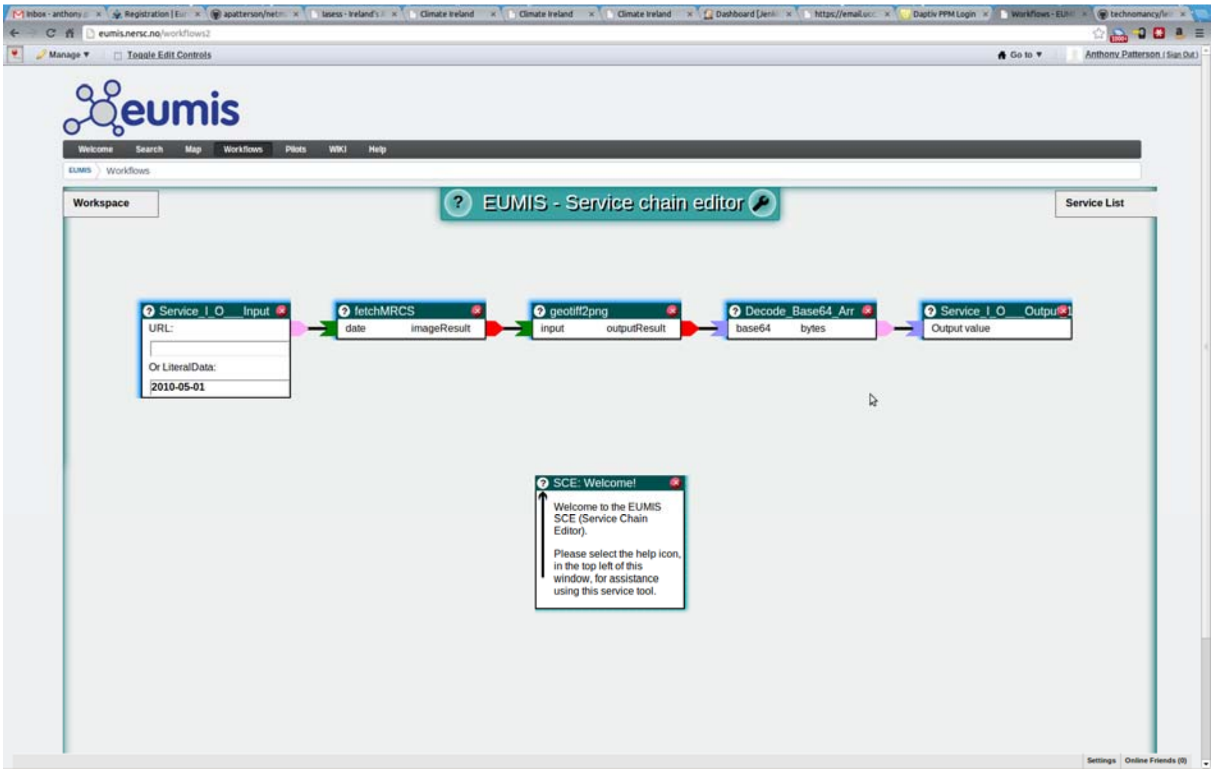


Figure 2-6 Chaining of web services in the service editor.

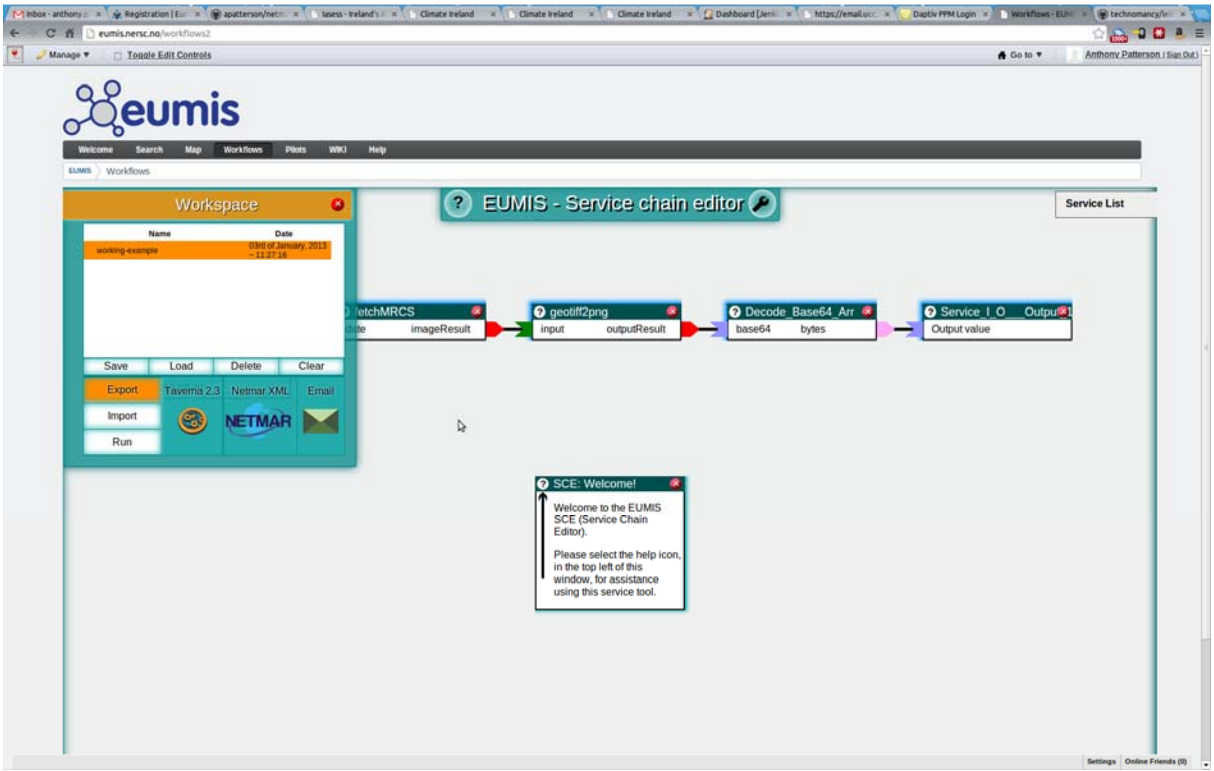


Figure 2-7 Exporting service chain

2.3.1.1.2 Run service chain

Once a user has defined and saved a service chain, he must be able to run it to verify that is produces the anticipated result. For this purpose, a service orchestration engine (or workflow engine) is needed.

An example of a service chain would be that a scientific user who has a satellite dataset containing, say, chlorophyll values at a particular resolution, would like to compare it with a model hindcast. The different processing steps would then be:

- The satellite dataset must first be converted to the same resolution as the model. If uncertainty information has been passed with the data then the interpolation routing used to resample the data should recalculate the uncertainty for the output and provide it to the next stage in the chain.
- The satellite dataset and model hindcast will be combined into a single output containing both sets of data for the intersection of the areas covered.
- The extracted data will then be passed to a comparison process which will carry out the statistical analysis as specified by the scientist. Uncertainty information passed down the chain will be used, for instance to calculate another data layer containing the difference in pixel values shown as a multiple of the per pixel standard deviation.
- The numerical data produce could, optionally, be passed to a web based visualisation process which would create plots and graphs of the statistics calculated in the previous step.

In the service orchestration engine, the user would open the service chain, select which satellite data and model hindcast to use as input data, and they press an Execute button to run the chain. The engine would then perform each of the steps above by calling the defined services and pass the output from one step on to the next step. When all processing steps have finished, the engine should pass the final result to a visualisation service where the user can generate diagrams and view them on the GIS viewer.

2.3.1.2 Components

Service chains will be built using a simple web based GUI editor. This will support a "drag and drop" style interface allowing the user to edit workflows with minimal training. Semantic tags will be used to present the user with suitable and compatible data and processing services.

Service chains will be saved using a simple XML representation and exported in the standard Taverna format (T2Flow) allowing third party tools to make use of them if desired.

Service chains will be executed within the portal by an independent orchestration engine based on Taverna Server. This will interface with the portal to run service chains saved by the editor portlet.

2.3.1.3 Rationale

A web based GUI approach was chosen for the service chaining editor as this is the least technically intimidating for users who have not worked with this type of technology before. Taverna was chosen as the basis of the service orchestration engine as it "offers a dataflow model and a more compact set of primitives that facilitates dataflow modelling and pipelined execution" [TAN10] compared to BPEL/BPMN based systems which provide a wider range of functionality but at the expense of greater complexity. Using Taverna as the orchestration system means that NETMAR workflows can be published on external sites and used by the wider community outside of the EUMIS portal.

3 Architectural Views

"A concrete architecture is an instantiation of a reference architecture achieved by substitution of the general, logical, abstract elements of the template with concrete or physical realizations by vendor products and instances of technical products, standards, protocols, and design/architectural decisions." [EGF+10]

In this section we describe the standards, products, protocols and decisions which are used to instantiate the RM-ODP reference model in order to provide concrete architecture elements which will drive NETMAR's implementation.

3.1 Architecture Overview

NETMAR's architecture design is driven by service-oriented architecture (SOA) design principles, and based on open standards from the Internet Engineering Task Force (IETF), World Wide Web Consortium (W3C), Organization for the Advancement of Structured Information Standards (OASIS), International Organization for Standardization/Technical Committee 211 (ISO/TC 211), and Open Geospatial Consortium, Inc. (OGC).

Figure 3-1 illustrates the main components of the NETMAR architecture. The portal supports applications including GIS map viewer, smart search and discovery client, and service chaining editor. These application components rely on the underlying service-oriented architecture, where there are seven main service components. Also depicted are the main flows of control, both planned and existing. The red and green lines depict what was implemented (green) and what was planned (red) at the time of publication of the Final Architectural Specification [PHD+12].

1. **Discovery Service:** data and service discovery service using CSW standard.
2. **View Service**: data view service using WMS standard.
3. **Download Service**: data download service using WFS, WCS, SWE, and OPeNDAP standards.
4. **Processing Service:** data processing service using WPS standard.
5. **Orchestration Engine:** workflow engine service (invoked via direct RESTful interface wrapped in a WPS).
6. **Semantic Web Services**: Including a smart search service using bespoke CSWM (CSW Mediator), an extension of the CSW to support smart search and an interface for retrieving knowledge from the NETMAR ontologies.

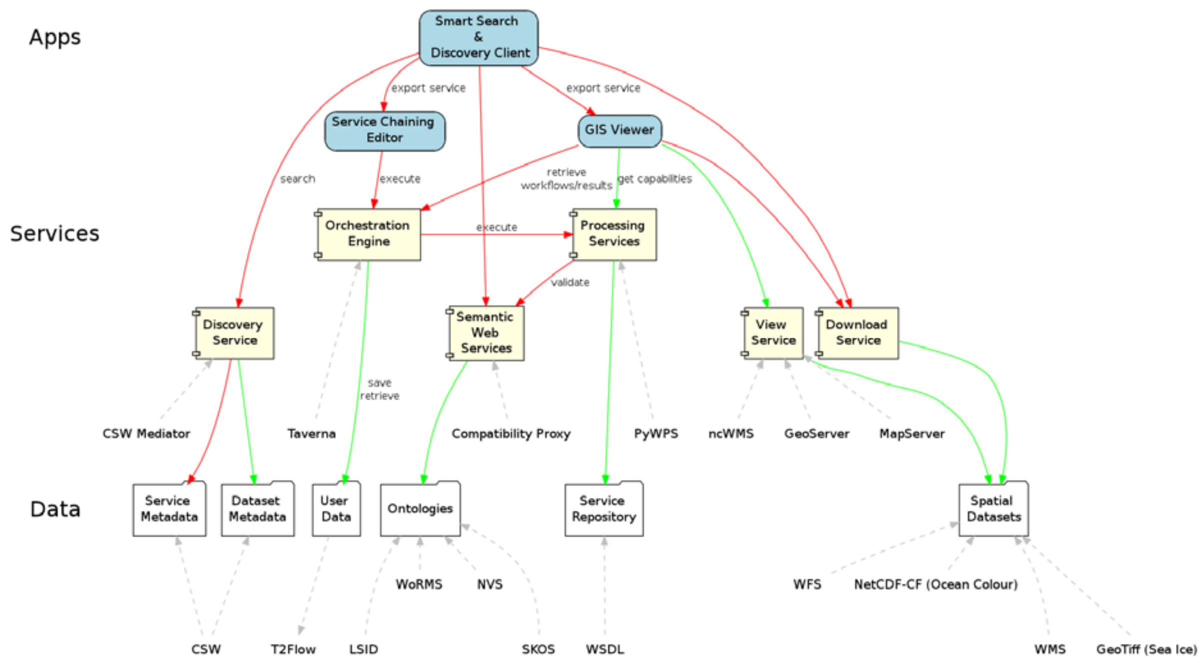


Figure 3-1 NETMAR Service Oriented Architecture

3.2 RM-OA Views

NETMAR follows an Architectural Views approach in which we describe the architecture from a number of complementary standpoints. The views chosen are based on the adaptation for Orchestra [OGC07] of the RM-ODP [ISO10] reference architecture (RM-OA).

- The Enterprise View is covered in detail in Section 4 and is concerned with the drivers of the project. These drivers are considered from the point of view of a number of stakeholders; researchers, end users and service providers.
- The Services (Computational) View is covered in Section 5 and is concerned with how NETMAR's use cases and requirements are resolved by the application of service oriented principles through the use of service oriented architecture patterns.
- The Information View in Section 3.4 shows the principal elements of data and metadata that allow NETMAR to satisfy its use cases, and shows how this information flows through the system.
- The Engineering View in Section 3.5 concentrates on the practical considerations of distributing services across partners' server infrastructures and how levels of service are maintained through monitoring and failover mechanisms.
- The Technology View (Section 3.6) lists the technologies, tools and standards used to build the NETMAR service oriented architecture.

3.3 Validation of Views approach

RM-ODP in its full implementation is a highly formal approach with its own specialised vocabulary and a long list of rules for its implementation. While RM-ODP is mentioned as being used in many of the projects related to NETMAR [DAP+10], none of them provide a fully conformant RM-ODP specification, but instead categorise their architecture into views roughly following the RM-ODP specification as is done here. While an initial attempt was done to provide a fully conformant RM-ODP architecture, this resulted in an initial version of the system architecture which did not adequately convey the concrete decisions needed to drive the implementation. This was at least partially due to RM-ODP not providing a suitable level of abstraction that allowed these decisions to be described clearly for an audience not familiar with UML and the concepts of reference architectures.

Subsequently, the requirement for RM-ODP compliance was dropped and a less formal approach, which focused more on patterns and practices in an overall view-based structure, was adopted. This proved to be far better suited to the task of describing NETMAR's architecture in a manner accessible to all of its stakeholders.

Recommendation: In order to describe the architecture of an environmental system in a way that is understandable across multiple disciplines, focus on patterns, principles and practices rather than formal descriptions and frameworks.

3.4 Information View

3.4.1 *Metadata*

The metadata invariant schema (Figure 3-2) constrains the way in which the NETMAR data providers supply metadata records in support of data and services. (Invariant Schema is a term from RM-ODP which indicates a set of relationships which must always be true for all valid behaviour of the system.) It does this in order that the implementers of the system may make reasonable assumptions about what metadata is available. It forms part of the contract between NETMAR and the providers of data to NETMAR.

The following are examples of the types of constraint to be imposed by the NETMAR metadata invariant schema:

- Metadata element names are bound to a unique meaning and type specified in ISO 19115 or 19119.
- Metadata values are bound to a published element name (and therefore to a meaning and type).
- A data or process service must have an associated metadata record.
- A dataset and/or a dataset series must have an associated metadata record(s).
- A data feature may optionally have a metadata record.
- The set of metadata elements used for semantic discovery and chaining is a subset of the published metadata records.

The circled "i"s within the diagram indicate Information Objects as defined within RM-ODP. The terms within braces are mandatory constraints on relations.

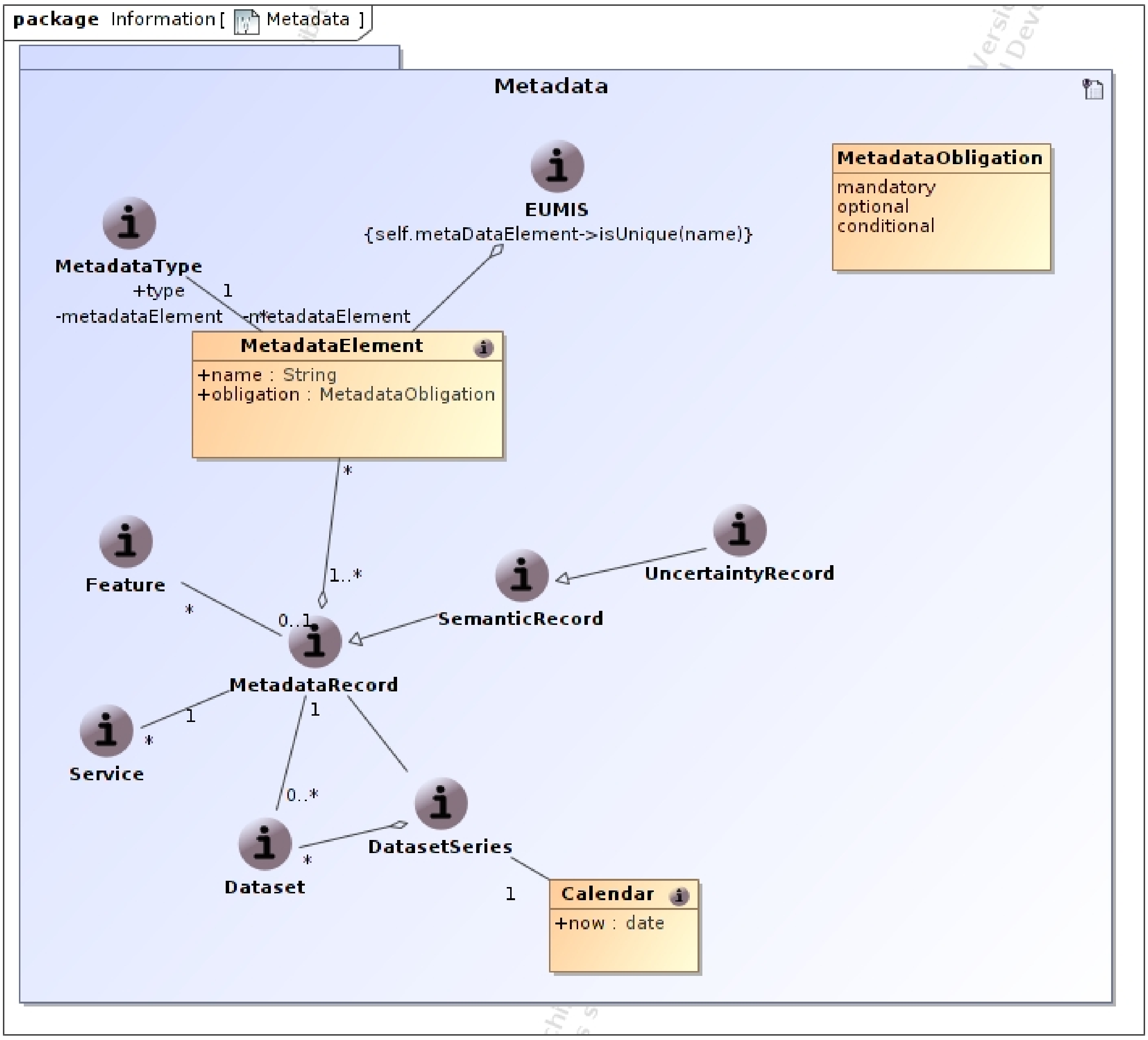


Figure 3-2 Metadata schema

The terms dataset and dataset series have the following meanings in this context:

- Data products that have determinate start and end times are defined as a dataset. This includes discrete data segment products sourced from continuously sensed data streams.
- Continuously sensed data products are defined as a dataset with an indeterminate end date.
- A dataset series is defined as a collection of datasets of the same product description, i.e. share similar characteristics. The exact definition is determined by the data provider. Examples include: a physical ocean model with same characteristics (e.g. same model processor, fixed spatial extent, etc.) but differing start and end times because of daily model run; a collection of individual raster maps captured from a common series of paper maps with differing spatial extent, etc.

The Semantics Invariant Schema (Figure 3-3) is intended to meet the objectives of the Semantics use-cases, e.g. Ontology Term Search (section 2.2.1.2.2, "Smart search”). It outlines a set of Information Objects related to the Artefacts outlined in the ontology term search Process.

A Term in this schema may be linked with another Term by a TermRelation. This is indicated in the diagram by an aggregation relation with cardinality 2, i.e., each TermRelation relates exactly two terms.Terms and TermRelations may be combined together in a TermGraph. This is shown by an aggregation relation going from TermGraph to Term, and another going from TermGraph to TermRelation. A Term has an associated Type and Description.

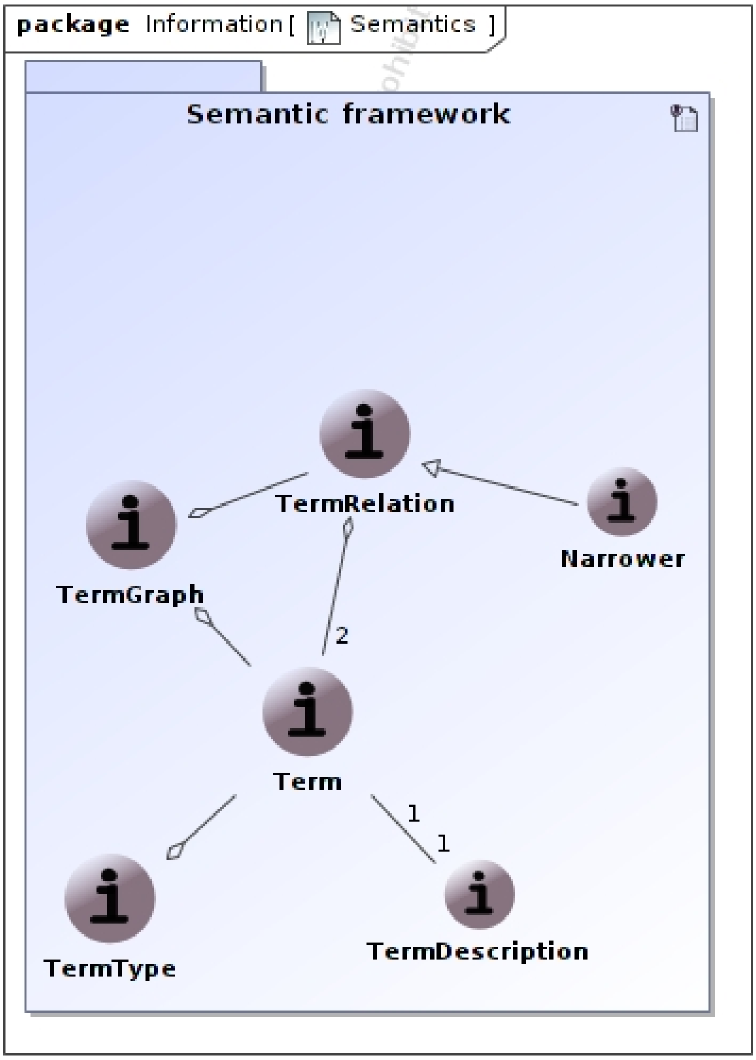


Figure 3-3 Invariant Schema for Semantic Framework

The set of metadata elements used for uncertainty propagation is a subset of the set of metadata elements used for semantic discovery and chaining.

To be able to estimate the reliability of the end result of chaining two or more processes together, we need a mechanism to convey the uncertainty or error data associated with each input, and to combine these in a meaningful way which can be attached to the output. Figure 3-4 shows how we might possibly create a link between a variable and its variance using a netCDF ancillary variable. A composition of these links comprises an uncertainty record associated with an input to a WPS process. The composition of uncertainties through a mathematical process itself results in an uncertainty associated with the WPS output. An example encoding using the UncertML netCDF binding is outlined in NETMAR D4.3.1 (NETMAR Semantic Framework Specification).

NETMAR has worked closely with the UncertWeb project in developing simple backwards compatible methods for encoding uncertainty in NetCDF files. A discussion paper on "netCDF-U" [BN11] has been produced by the UncertWeb project and this is compatible with the NETMAR approach.

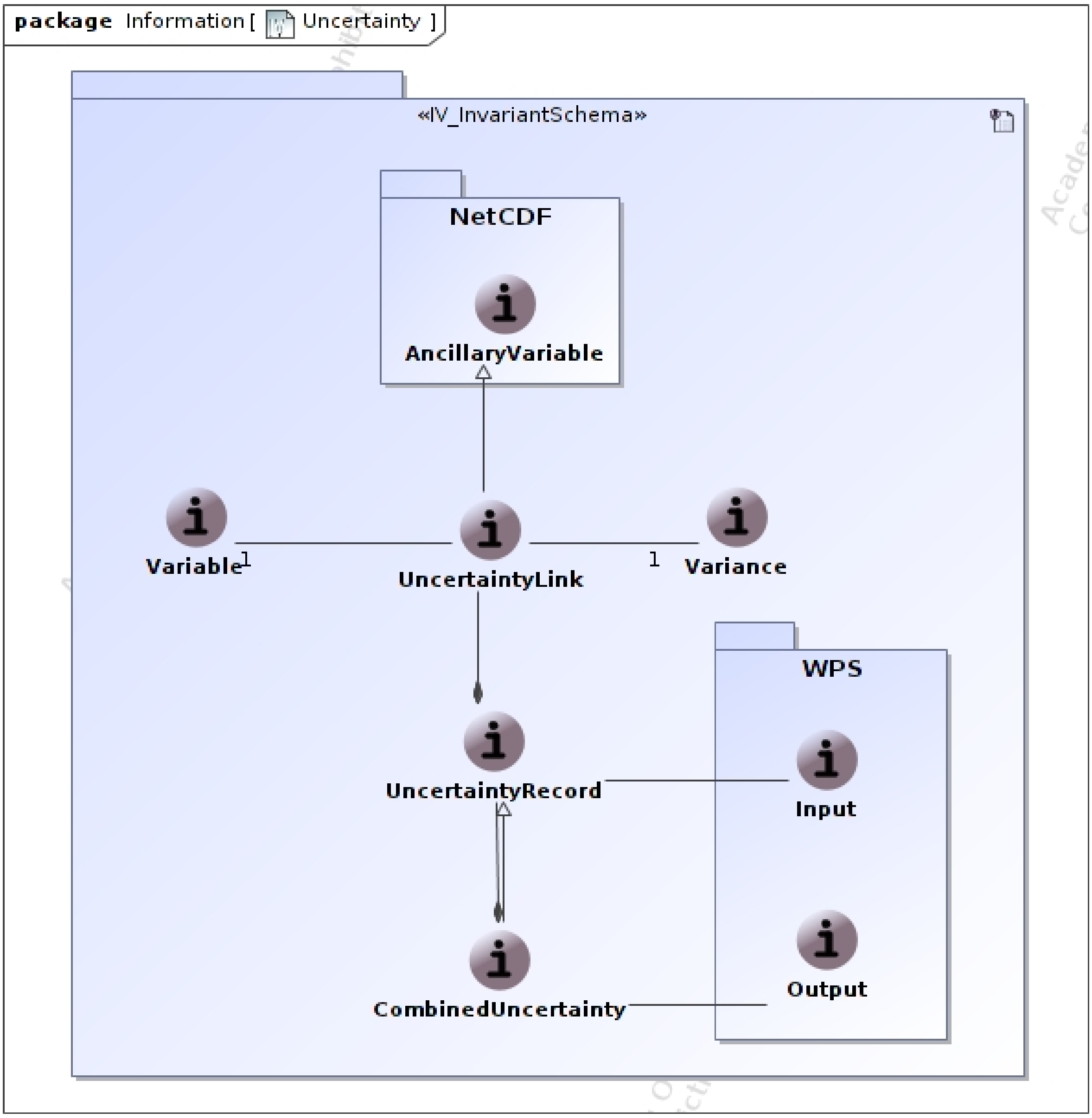


Figure 3-4 Uncertainty metadata

Finally, a metadata profile specification is required by the NETMAR system. NETMAR Core is the minimum set of metadata elements that should be supported by anyone wishing to publish data on the NETMAR platform. It is an ISO19115 profile.

The NETMAR metadata profile is documented in Appendix A.

3.4.2 *Information Flow*

The expected results of the Ocean Colour – Marine Ecosystem, Research and Monitoring pilot include making interoperable datasets and online comparison tools available to a wide audience to facilitate the use of advanced data analysis and statistics to support scientific studies.

The end user requests, through the portal, one of a number of possible statistical analyses of a combined dataset. The Orchestration Server hides from this user the details of how the datasets are combined. Under the hood, the Orchestration server combines the results of a chain of data services and processing services to produce the end result. For instance, the satellite data must undergo a change in resolution before being combined with the model data and passed to the statistical analysis services. The result is then passed to the Data Scientist as a graphical image which appears in the appropriate window of the portal (Figure 3-5).

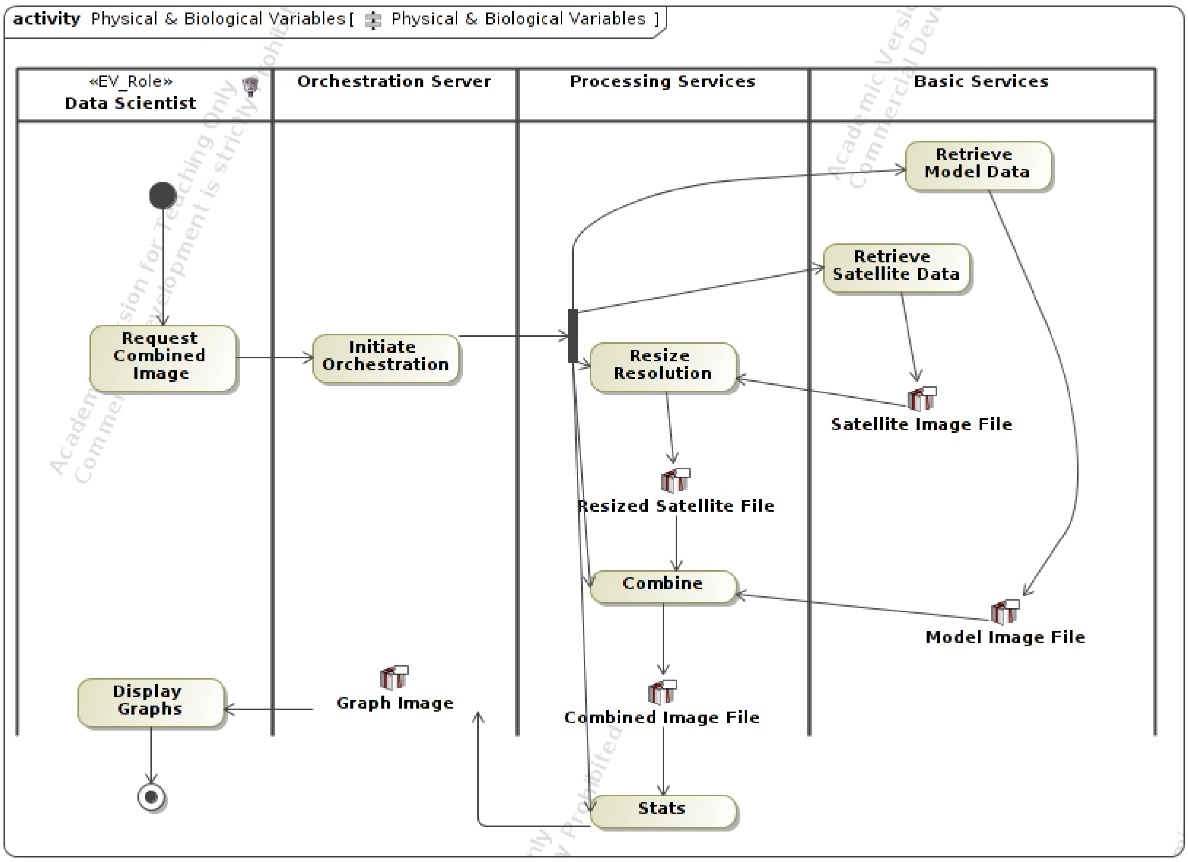


Figure 3-5 Combining Satellite and Model Data

3.5 *Engineering View*

3.5.1 *Monitoring and Availability*

As many of NETMAR's data and service providers are already running operational systems, a variety of strategies are already in place for monitoring service levels and ensuring service availability. In a straw poll of partners, the majority of respondents had some automated monitoring of quality of service, with Nagios being the preferred solution. Some had coupled this with automated failover and restart of critical components. One partner was in the process of moving to a fully virtualised environment with high availability and dynamic load balancing built in.

3.5.2 *Operational characteristics of the data*

The fact that NETMAR deals mainly with reading data and does not have many operations which involving clients altering state on a service simplifies some aspects of data management. Very few partners saw a need for transaction management. This would make sense: if state is not being altered while data is being processed, then there is nothing to rollback should the transaction fail. A read failure during any part of a process chain should simply result in no results being returned, without any particular risk of the system being left in an inconsistent state. No partners reported the presence of non-idempotent operations in their service interfaces, so service calls which fail to return after a specified timeout can be repeated without undesirable side effects. We define side-effects from an end-user perspective, i.e. no state was modified which the user cares about, so we do not include non end-user visible activities such as logging.

As well as improving system robustness, idempotency can improve the understandability of a system by reducing the need to reason about internal state. This is particularly important in a system such as NETMAR, which is intended to allow users who are not familiar with programming techniques to construct chains of services.

Recommendation: Designers of services should strive to avoid the need for complicated transaction management by ensuring, where possible, that repeated invocations of the same service request have the same user-visible consequences.

The large amounts of data handled could be expected to create problems with bandwidth, and computation. Some partners mentioned using caching strategies to reduce problems with bandwidth. Only one partner reported using a grid strategy for computationally intensive tasks. This was reported as being a batch process running manually rather than something which could be easily pipelined.

3.6 Technology View

Table 3-1 lists the technologies used to build the NETMAR service oriented architecture and its service network. The discovery, view, download and processing services all use standardised and well-established protocols. Standardised service oriented protocols for the orchestration engine service, smart search service, and semantic service do not currently exist and must be bespoke or custom-built based upon underpinning W3C standards. However, the orchestration engine and smart search services will be specifically based on WPS and CSW protocols respectively, and can therefore be regarded as special profiles or extensions of these existing standards. It is also planned that the NETMAR SWS (Semantic Web Service) specification will be submitted to the GEOSS Best Practices Wiki (http://wiki.ieee-earth.org).

Name	Protocol	Tools
Discovery Service	CSW	GeoNetwork
View Service	WMS	GeoServer, MapServer, ncWMS, THREDDS (ncWMS)
Download Service	WFS	GeoServer
	WCS	GeoServer, MapServer, THREDDS
	SWE	Constellation
	OPeNDAP	THREDDS, Dapper
Processing Service	WPS	PyWPS
Orchestration Engine Service	bespoke (WPS wrapper)	Taverna server (with SCUFL2 XML workflow descriptions)
Smart Search Service	bespoke (CSW Mediator)	Custom built using Java

Semantic Service	bespoke (NETMAR SWS specification)	Custom built using Java, Jena, AllegroGraph
NERC Vocabulary Server	SKOS, DCMI	Oracle, Custom built using Java

Table 3-1 NETMAR service network technology list

Table 3-2 lists the technology components of the NETMAR portal.

Name	Tools
Portal framework	Liferay
WebGIS client	Mapfaces / Openlayers / jQuery
Smart search & discovery client	Adobe Flex
Service Chaining Editor	jQuery

Table 3-2 NETMAR portal components technology list

3.6.1 *Cookbooks*

NETMAR has produced a set of detailed guidelines introducing the technologies used in NETMAR. These are intended for a general audience and are written to allow users to quickly get up to speed on how the NETMAR components, standards and technologies are used in the context of NETMAR.

The cookbooks cover the following topics:

- Establishing standard data services, suitable for use with NETMAR and the wider GEOSS common architecture.
- Establishing standard processing services with PyWPS, suitable for use with NETMAR and the wider GEOSS common architecture.
- Adding semantic annotation to standard OGC datasets and processing services.
- Adding uncertainty to datasets and processing services in an OGC environment.
- ICAN semantic interoperability pilot cookbooks:
  - Understanding metadata.
  - How to establish a standard CSW metadata catalogue with GeoNetwork.
  - Understanding semantics.
  - How to connect my coastal atlas to the ICAN semantic interoperability pilot.

4 Analysis

The Enterprise View is concerned with the drivers of the project. These drivers are considered from the point of view of a number of stakeholders;

- The community of researchers and practitioners who are interested in the lessons learned from NETMAR and how they may apply them to their own projects,
- The end users of the NETMAR prototype who are interested in seeing a product which can make their day-to-day tasks easier,
- The community of service providers, data providers and consultants who are interested in using NETMAR as a platform for adding value to its existing data and services.

4.1 *Research Drivers*

4.1.1 *Constructive Research*

NETMAR has as a strong central objective, the production of advice on good practices to follow when constructing an environmental system of systems. In order for that advice to be credible, NETMAR will attempt to follow a research method common in software engineering called Constructive Research.

[Shaw01] describes a number of research settings. The Method/Means setting asks the questions “How can we accomplish X? What is a better way to accomplish X? How can I automate doing X”. This can be tackled by applying the research strategy “Build a Y” and the validation strategy “Measure Y, compare to X”.

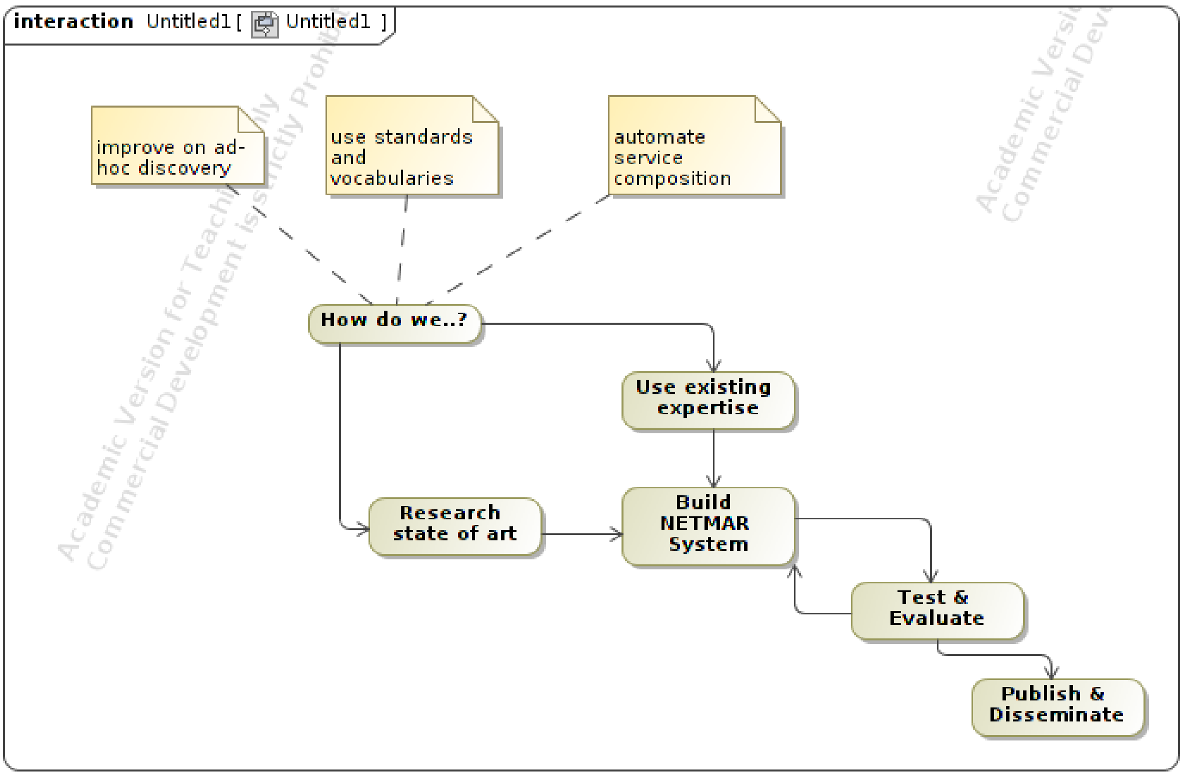


Figure 4-1 Constructive Research

To apply this approach to NETMAR, we can ask a number of questions along the lines of

- “How can we improve on ad-hoc discovery of datasets and services?”

- "Will having a unified interface to services help researchers?"
- "Is a web-based graphical interface a more accessible way of composing services than through scripting?"
- "Can standards and controlled vocabularies help to solve scientific problems?"

To answer these questions we build the NETMAR system (the Construct), and measure how well the system performs vis-à-vis the status quo. To do this we engage a set of real world stakeholders who describe various scenarios in which they discover data, perform calculations, combine with other data, and publish results. The stakeholders are selected from a number of user groups, both operational and research based. Having built a system which attempts to improve the way they do their work, we conduct a set of evaluations where we measure their impressions of how using the NETMAR system compares to their current toolsets.

This objective of providing usable research results comes into play when making architecture choices. For example a single sign-on strategy would have obvious benefits for the end user in a system such as this where data and services are provided by different providers with different access policies, but as it does not directly affect the research questions we asked, we assign it a low priority for NETMAR. We must be careful though, that in constructing an idealised solution where we may make certain simplifying assumptions such as the above, we do not limit the applicability of the idealised solution to the real world. In the example given, we can design the system assuming that all datasets and services are freely available, even though we know that this is not always true in the real world, because we know that single sign-on systems are available and implementable. We can say that "if a user has all the credentials necessary to access the datasets and services, then that will be equivalent, in implementation terms, to the case where the data is freely available". We must also be careful not to compromise the usability of the system to the point where it becomes unusable, as that will affect our ability to validate results.

In the example above, 'freely' is intended in its most liberal sense, i.e. there are no restrictions on accessing the data, not even the requirement on the part of the service accessing the data to identify itself. In most real-world situations more stringent requirements will exist and the above assumption will not be true. Other projects, e.g. MashMyData (www.mashmydata.org/) are addressing these issues in detail. NETMAR will monitor progress in this area and allow room for expansion e.g. by allowing specific service metadata records which could incorporate access and security policy details.

A similar argument pertains to data licensing. We assume that a policy or technology based solution can be found to the problem of heterogeneous licenses and therefore do not make solving this a priority for NETMAR.

4.1.1.1 Validation

NETMAR's conclusions would have been stronger if users had had a better chance to evaluate the product without unnecessary distractions. The evaluation reports [T+11] refer to difficulties users encountered with users the NETMAR system because of specific instances of functionality not working, or working incorrectly. A lot of these bugs could have been avoided by the application of industry standard software engineering practices. Frequently, changes being made in the application code resulted in errors in the production environment. A simple branching and merging strategy with release management could have kept development and production environments separate and avoided many of these cases. Another standard practice is to use standard tools to automate build and testing. In combination with frequent code releases in order to continuously check for integration errors, these techniques have been employed with success on projects of all sizes, even research projects[WK03], to ensure that end-users have at all times a viable product available for testing.

Recommendation: Industry standard software engineering practices should be applied on all projects, including pilots, in order to ensure a minimum level of product quality.

4.1.2 *Patterns*

Another research driver is the desire to use and acknowledge prior art. NETMAR's decision to use service oriented architecture principles and practices is an acknowledgement that there exists a body of knowledge related to the types of problem NETMAR is solving and that it would be wise to build upon this rather than trying to reinvent the wheel. Section 4.80 lists a series of projects and initiatives from which NETMAR's architecture has drawn inspiration but is not an exhaustive list. A fuller list is available in [DAP+10].

Enumerating all of the prior projects which NETMAR draws upon would not be possible. Fortunately there is a way of drawing on the experience of multiple projects without needing to know the full details of the projects themselves. Patterns and patterns languages are an idea borrowed from the architecture of buildings which facilitate design reuse [Bus96]. A view of service oriented architecture that has become quite prevalent in the last few years is of a catalogue of patterns from which the software architect can choose in order to build a system whose design promotes SOA principles. [RBD11] is one such catalogue upon which section 5 of this report relies heavily.

Patterns are an architectural driver in the sense that they exert a strong influence on how we design the system. In the case where a pattern exists which is a reasonably good fit in our solution space, we will favour this over designing a solution from scratch, even though the custom solution might be better in some ways. This is because the pattern allows us to learn from others' experience which quality attributes the solution affects, what its potential tradeoffs and risks are, and how it interacts with other patterns.

4.1.3 *Validation*

The results of applying individual patterns are documented in Section 5 Service Patterns. The decision to embrace the patterns approach is validated by the review comments on the Final Architecture Report which were much improved on the initial report. The reviewer was able to gain confidence in the decisions made based on their description in the report. The description of the interactions of the system's components in terms of simple patterns was followed by an increased understanding and new ideas for how the services could interact in new and useful ways, e.g. the semantic validation component

Recommendation: Patterns for a simple basis for design reuse and as such should be considered on every project as a way of learning from the successes and mistakes of others.

4.2 End User Drivers

NETMAR's user visible functionality is based on extensive interviews with end users based on the four NETMAR pilots. This process and the resulting use cases and requirements are documented in [PTS+11]. In addition the system has been tested extensively to obtain user feedback, resulting in the addition of new requirements [T+11]. Many of the end user requirements are directly related to areas of functionality covered under NETMAR's core capabilities. For example, support for common GIS operations, such as panning, zooming, selecting layers, etc. comes as part of the use of standard components based on OGC services provided through a common portal. NETMAR has a role to play both in evaluating how relevant these features are to our users' needs and soliciting new requirements for future enhancements.

4.3 Value Add Drivers

Although NETMAR will be necessarily limited in scope in terms of its overall functionality, it is the intention of the NETMAR consortium to build a community of service providers around a common platform of managed services, openly available tools and a defined governance process.

NETMAR's codebase will be made available under an open source license for use by other projects. This however will not necessarily be enough to ensure that NETMAR's usefulness can extend beyond the project's lifespan, so a series of cookbooks has been published showing how this code can be put to good use in the context of other projects.

An illustrative scenario arose during the processing of the feedback from the first round of user testing. It emerged that some users would like the process chaining to be triggered by a event, rather than manually by the user. Although the workflow architecture permits this, it would require extra work to implement. Rather than spending additional resources implementing a new feature requested by one pilot, it was felt that there would be more value in documenting this as an example of how a third party could extend a part of the NETMAR system to meet a new use case.

4.4 Traceability

The following table is a guide to how the high-level requirements, as derived from the initial pilots and the subsequent feedback from user testing, relate to different sections within this architecture document.

Id	*Description*	*Reference*
R1	All data sources must provide usage metadata necessary to automatically combine different sources: • Projection used. • The units of measure used for each parameter using a term from a defined vocabulary. • Each parameter should be labeled with a term from a defined vocabulary.	Usage metadata
R2	All data sources must provide data using standard interfaces.	Technology View
R3	When a new relevant data source becomes available, it should be possible to include it in the service without any additional programming as long as it follows the required data protocols and usage metadata.	Information View Metadata Technology View Value Add Drivers
R5	Maps shall be served via WMS so that they are viewable in several different GIS viewers.	Technology View
R6	Visualization of parameters shall be done so they give relevant information to a end user. E.g. use symbols that are familiar to the end user.	End User Drivers

Id	*Description*	*Reference*
R7	The EUMIS portal must support inputting a route that shall be displayed as a part of a generated map.	End User Drivers
R8	The EUMIS portal must support selecting a specific area of interest.	End User Drivers
R9	The EUMIS portal must support sharing defined service chains with other users .	End User Drivers
R10	The EUMIS portal must support creating, saving and loading service chains.	End User Drivers
R11	All data sources must provide data using standard interfaces so that the information from several sources can be combined. - Allows to browse/search resources	Technology View
R12	Extended tools for changing the content of the map module, e.g. to change the displayed variables, or adjust the time.	End User Drivers
R13	Display different model results layers in order to compare them with each other Display all relevant data (meteo-oceanic data …)	End User Drivers
R14	Display synthetic model results layers Access to synthetic maps and information	End User Drivers
R15	When new relevant data sources (pollution observation, model results, model comparison analysis ..) become available it should be possible to include it in the service and send an alert to end-user	Value Add Drivers
R16	The EUMIS portal must Allows to navigate within scenario and/or time steps	End User Drivers
R17	The EUMIS portal must allow context-dependent actions such as publishing/un-publishing	
R18	The EUMIS portal must give access to cleanup history of a designated site The EUMIS portal must display all relevant data (shoreline pollution, shoreline sensitivity, …)	
R19	Enables to view features displayed as charts (for shoreline and cleanup survey evolution and statistics) Access to synthetic maps and information	
R20	All processing services must provide usage metadata sufficient to allow data sources to be matched to processing inputs.	Usage metadata
R21	The EUMIS portal must support running predefined service chains.	Run service chain
R22	Statistical comparison and generic web processing services are available.	Service Host

Id	*Description*	*Reference*
R23	Datasets and processing services should support uncertainty throughout the processing chain.	**4.5 Information View** Metadata
R24	The creation of new service chains using data services that provide the necessary metadata and data protocols, without complex programming, should be supported by the service chaining editor within the portal.	Value Add Drivers Service chaining editor and orchestration engine
R25	Data sources must provide at least discovery metadata.	**4.6 Information View** Metadata
R26	All discovery metadata should be formatted and delivered through open standards.	Technology View
R27	All accessible data sources must use open standards and open data transport formats. Any restricted data sources must describe means of obtaining data within the metadata (e.g. link to registration page, email address, etc.).	End User Drivers
R28	EUMIS system needs to facilitate regional base metadata profile standards (e.g. INSPIRE in Europe, FGDC in U.S., etc.).	Appendix A. NETMAR Discovery Metadata Profile
R29	Support for metadata visualisation.	
R30	Support for smart search functionality, i.e. ability to search a keyword based on semantics.	Smart search
R31	Support for multi-lingual and multi-domain ontologies.	
R32	Support for ontology browsing.	Ontology browsing
R33	The EUMIS portal must support common GIS operations like zoom, pan, restore original map extent, toggle layer display on/off, and simple colour manipulation.	End User Drivers
R35	Support display of legends in web-GIS.	End User Drivers
R36	User interactions should be as intuitive as possible, and allow the user to explore the available datasets as effortlessly as possible.	End User Drivers

Id	*Description*	*Reference*
R37	Any user must be able to search the wiki and forum for information about products and services offered by EUMIS. The search criteria can be free text.	Composite Front End [RBD11]
R38	Any user must be able to navigate all pages of the wiki, and quickly get back to previous pages by using the "breadcrumb trail" at the top of the screen or the browser's back button.	Composite Front End [RBD11]
R43	A service provider must be able to enter content into the wiki pages to describe his products and/or services.	Value Add Drivers
R44	The administrator must be able to assign permissions for editing wiki to a new provider/user, and place him in one of the defined communities.	Composite Front End [RBD11]
R45	The administrator must be able to reject a user that provides an invalid e-mail address.	Composite Front End [RBD11]
R47	The EUMIS portal must support display of multiple parameters in a common map projection, even if they have different spatial resolution and/or map projection.	
R48	The EUMIS portal must support "smart search" by parameter name, allowing also datasets with related parameters to be found.	Smart search
R49	The EUMIS portal must support queries by parameter name ("smart search") combined with geographic area and/or time range.	Smart search
R50	The GIS viewer must support displaying data in polar stereographic projection since this is the most useful projection for data in the Arctic.	End User Drivers

4.7 Comparative Analysis

A number of initiatives, standards and technologies have already been identified as being relevant to NETMAR's architecture [DAP+10]. We compare a selection of these with NETMAR with the objective of finding potential points for information sharing.

4.7.1 *INSPIRE*

An expected impact of NETMAR is to support the implementation of the INSPIRE Directive. It is therefore required that the NETMAR architecture specification dovetail with INSPIRE's overall architecture, common principles, implementation rules and guidance documents. An overview of the INSPIRE architecture is illustrated in Figure 4-2. The Directive requires that common implementation rules are adopted in a number of specific areas (i.e. metadata, data specifications, network services, data and service sharing, and monitoring and reporting). NETMAR as an ICT project is specifically focusing on the areas of metadata, data specifications, and network services.

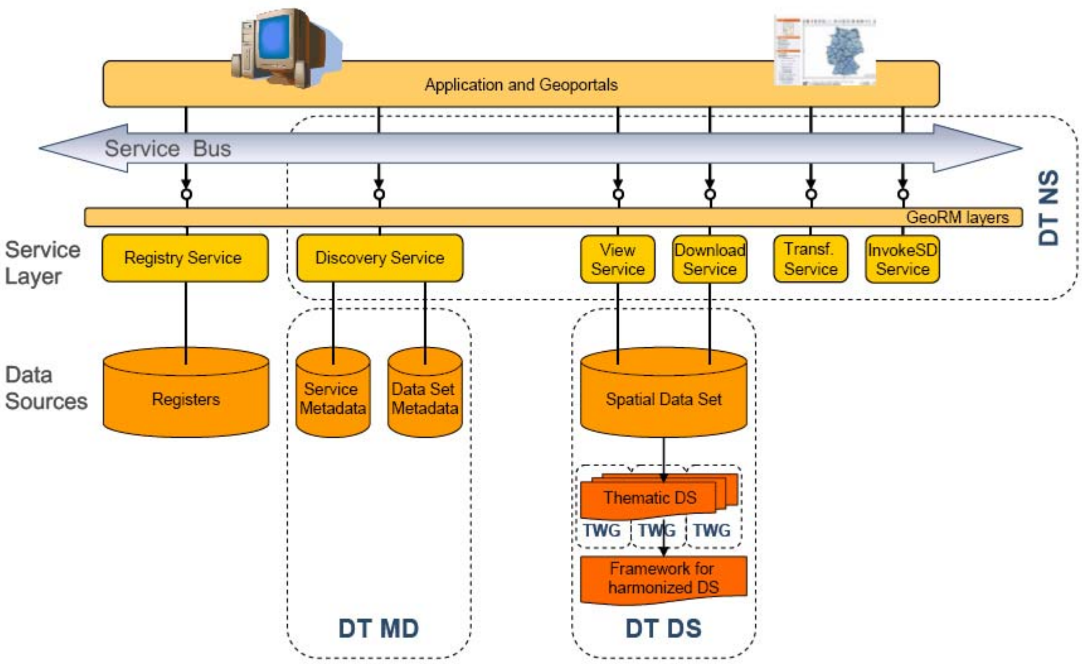


Figure 4-2 INSPIRE architecture [INS07b]

To assist implementation, technical guidance documents are provided by INSPIRE. These documents and related supporting materials give detailed technical documentation concerning the relevant implementation rules. Table 4-3 lists the current (January 2013) status of relevant INSPIRE guidance documents. As INSPIRE guidance documents are updated during the course of the NETMAR project, it is also necessary to incorporate these changes into the NETMAR system architecture if and as appropriate.

Implementation Rule	Guidance document	Status note	Version	Publication date
Metadata	Technical Guidelines based on ISO 19115 and ISO 19119	-	V1.2	16-06-2010
Data Specifications	Annex I Data Specifications (9 documents)	-	v3.0.1 to v3.1	03-05-2010
	Annex II Data Specifications (4 documents)	Draft	v3.0 rc2	06-07-2012 to 09-07-2012
	Annex III Data Specifications (19 documents)	Draft	v3.0 rc2	30-04-2012 to 09-07-2012
Network Services	Discovery Services	-	v3.1	07-11-2011
	View Services	-	v3.1	07-11-2011
	Download Services	-	v3.0	12-06-2012
	Schema Transformation Service	-	v3.0	15-12-2010
	Coordinate Transformation Services	Draft	v2.1	15-03-2010

Table 4-3 INSPIRE guidance documents (table updated January 2013)

4.7.2 *SEIS and SISE*

The Shared Environmental Information System for Europe (SEIS) was proposed by the European Commission in January 2008. It is a collaborative initiative of the European Commission and the European Environment Agency (EEA) to establish with the Member

States an integrated and shared pan-European environmental information system. The SEIS concept is based on a decentralised system, but integrated and web-enabled. It aims to move away from fragmented reporting systems to online information systems where Member States create services to make information available to multiple users (Figure 4-3). Such online information systems will better integrate all existing data gathering and information flows related to EU environmental policies and legislation. A major challenge is to organise the large amount of environmental data and information and to integrate these, where desirable, with existing social and economic data. This data should be made available together with tools that allow experts to do their own analyses and to communicate their results in ways which policy makers and the public can easily understand [SEISa]. It will build upon the experiences of initiates such as the INSPIRE Directive and EEA Reportnet (part of the EEA Eionet partnership network).

SEIS is based on the following principles [SEISb]:

- Manage information as close to its source as possible.
- Collect information once, and share and use often.
- Help public authorities with their legal environment reporting obligations.
- Help public authorities to assess the state of environment, and environmental policy effectiveness, and to design new environmental policy if needed.
- Help make geographical comparisons of the environment.
- Help citizens to participate in the development and implementation of environmental policy, while making information fully available to them.
- Based on open standards.

Since 2007, the EEA has visited many Member States and cooperating countries to explain SEIS, encourage implementation, and discover existing SEIS activities at the national level. Conclusions to date include [SEISb]:

- Some countries are fairly advanced in implementing SEIS principles, while others need to take significant steps.
- Most are up to date with the new opportunities offered by modern ICT.
- Some countries have a need for better cooperation between institutions.
- The benefit of having access to European information within a national context is still often vaguely recognised within countries.

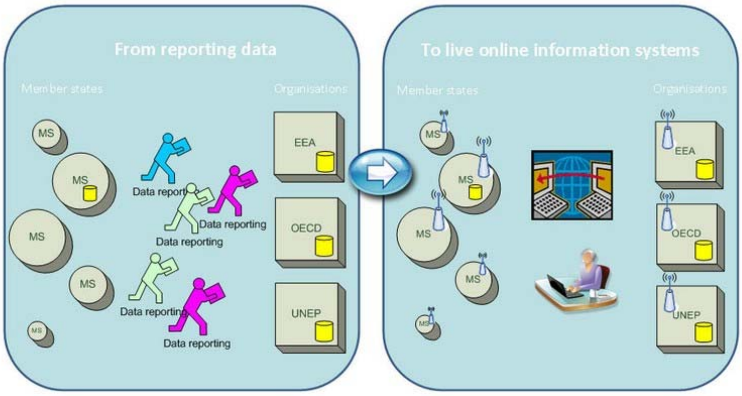


Figure 4-3 SEIS: from fragmented reporting systems to online information systems [SEISb]

Another initiative, the Single Information Space in Europe for the Environment (SISE), was proposed by the European Commission's DG Information Society and Media (INFSO) during the preparation of FP7 Work Programmes. The aim of SISE has an ICT research vision to enable [Hreb09]:

- real-time connectivity between multiple environmental resources;

- seamless cross-system search;
- cross-border, multi-scale, multi-disciplinary data acquisition, pooling and sharing;
- service-chaining on the web, thereby stimulating data integration into innovative value-added web services.

Topics such as service-oriented architecture (SOA), web services, process chaining and uncertainties, real-time mapping and modelling, controlled vocabularies, data interoperability, open standards, open source software, model driven communities, web communities, data visualisation and SISE deployment models are all central to SISE [SISE2].

Both the SEIS and SISE initiatives are seen as complementary. SEIS is taking a more top-down approach which is establishing a legal and policy basis for an integrated and sustainable EU-wide eReporting System. In contrast, SISE is taking a more bottom-up approach which is focused on ICT research to achieve better integration of environmental information systems. However, both initiatives rely on INSPIRE for providing the overarching Spatial Data Infrastructure (SDI) framework. Currently, there is no EU budget specifically dedicated to SEIS. However, an official implementation plan for SEIS is expected from the EC. To date, implementation of SEIS principles is supported through EC research programmes such as FP6 and FP7.

NETMAR is specifically funded by DG Information Society and Media (INFSO), where expected impact is to contribute to SISE through improved systems connectivity and semantic interoperability. NETMAR is liaising with relevant SISE cluster projects funded by INFSO such as UncertWeb (http://www.uncertweb.org), ENVISION (http://www.envision-project.eu), etc. to help achieve better leverage towards SISE. NETMAR is also expected to contribute to the development of SEIS where the outputs of such ICT research (e.g. system architecture) is seen as essential for supporting the implementation of SEIS.

4.7.3 *GENESIS and GIGAS*

The objective of the GENESIS project[2] was to develop efficient solutions for Environment management and Health actors, based on advanced ICT to constitute collaborative information networks and integrating existing systems. GENESIS used RM-ODP to describe the architecture of six thematic pilots with air and water quality:

- Air quality pilot in London (UK)
- Water Quality Pilot on Bodensee/Constance Lakes (Germany)
- Air quality pilot on Bavaria (Germany) [DLR09]
- Water quality pilot on Oder Estuary (Baltic Sea) [SGH09]
- Water quality pilot on Villerest Reservoir (France)
- Air and Coastal Water Quality Pilot on French Riviera [ACRI09]

While the application domains for the pilots are different from those of NETMAR, their system architecture descriptions provide valuable information on how to describe the architecture for a pilot EIS. The GENESIS architecture is comprised of an application and geoportal framework, a service bus and a set of services (user registration, discovery, data delivery, data processing) as illustrated in Figure 4-4. The GENESIS portal is realised by the Liferay portal framework[3], and contains the following system components:

- Portal Factory: Allows administrators to create and deploy thematic and personalized portals in Liferay, and users (with correct permissions) to customise these to suit their purposes.

[2] http://www.genesis-fp7.eu/

[3] http://www.liferay.com/

- Geodata Vizualisation: Offers a powerful and flexible WebGIS Client for displaying the various data sources (in situ, EO, GIS, non-geographic) that can be integrated in the GENESIS portal.
- Workflow & Service Orchestration GUI: Provides an easy to use graphical user interface for composing service chains of web services (e.g. WPS services).
- System Supervision & Admin GUI: includes facilities for user management, resource administration, etc.
- Workflow Factory: Offers and generates a testing interface for activating different workflows generated within the GENESIS environment.

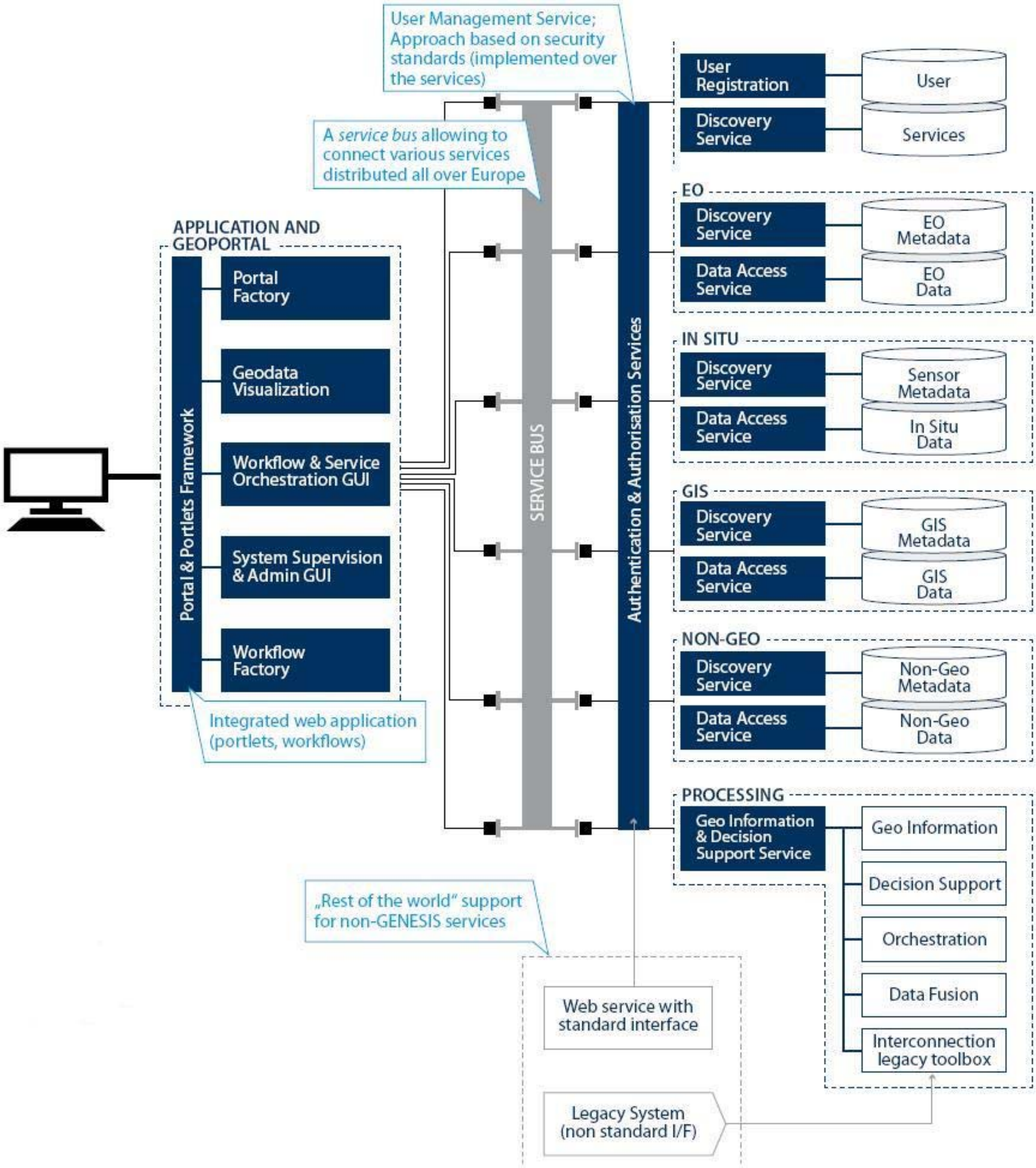


Figure 4-4 The GENESIS service oriented architecture [GEN11].

The GIGAS project[4] analysed the system architectures of INSPIRE, GMES and GEOSS using RM-ODP [GC09a] [GC09b] [GC09c] [GC09d] [GC09e]. These specifications provide detailed information about the system architectures of these key SDI initiatives and as such serve as reference information for NETMAR. The GIGAS project formally ended in May 2010, but the GIGAS Consortium initiated the European GEO SIF[5] to continue coordination of European requirements for GEO and GEOSS related activities. At the time of writing, these were no new material since the reports listed above, but any future updates from this initiative will be followed closely by NETMAR.

4.7.4 *W3C, OMG, and OASIS*

A number of web services and web GIS standards have been monitored for possible relevance to the NETMAR architecture. The following standards are found to be relevant:

- OGC data delivery services:
  - Web Map Service (WMS) [WMS13]
  - Web Feature Service (WFS) [WFS04]
  - Web Coverage Serivce (WCS) [WCS11]
  - Sensor Observation Service (SOS) [SOS06]
- OGC data processing services: WPS [WPS10]
- OGC metadata catalogue service (CSW) [CSW202]
- Open-source Project for a Network Data Access Protocol (OPeNDAP)[6]
- OGC Web Map Context (WMC) [WMC05]
- Simple Knowledge Organization System (SKOS) [MB09]
- Business Process Modeling Notation (BPMN) [OMG08]
- OASIS Reference Model for Service Oriented Architecture (SOA-RM) [OAS06]

Some of the standards initially reviewed are currently not used in NETMAR, but listed below for possible future inclusion:

- SoaML [OMG09]: A UML profile and meta-model designed to support service modelling and design in a model-driven development approach.
- Rules Interchange Format (RIF) [W3C10]: A W3C working group whose charter is to look at various rules languages with the aim of providing a standardised language for declarative rules and production rules.

[4] http://www.thegigasforum.eu/project/project.html

[5] http://www.thegigasforum.eu/sif

[6] http://www.opendap.org

5 Service Patterns

Patterns describe reusable elements of design in the form of narratives describing a problem with its solution and guidelines for applying it. In this section we describe NETMAR's application of service oriented architecture in terms of its usage of some documented SOA patterns.

5.1 Composite Front End [RBD11]

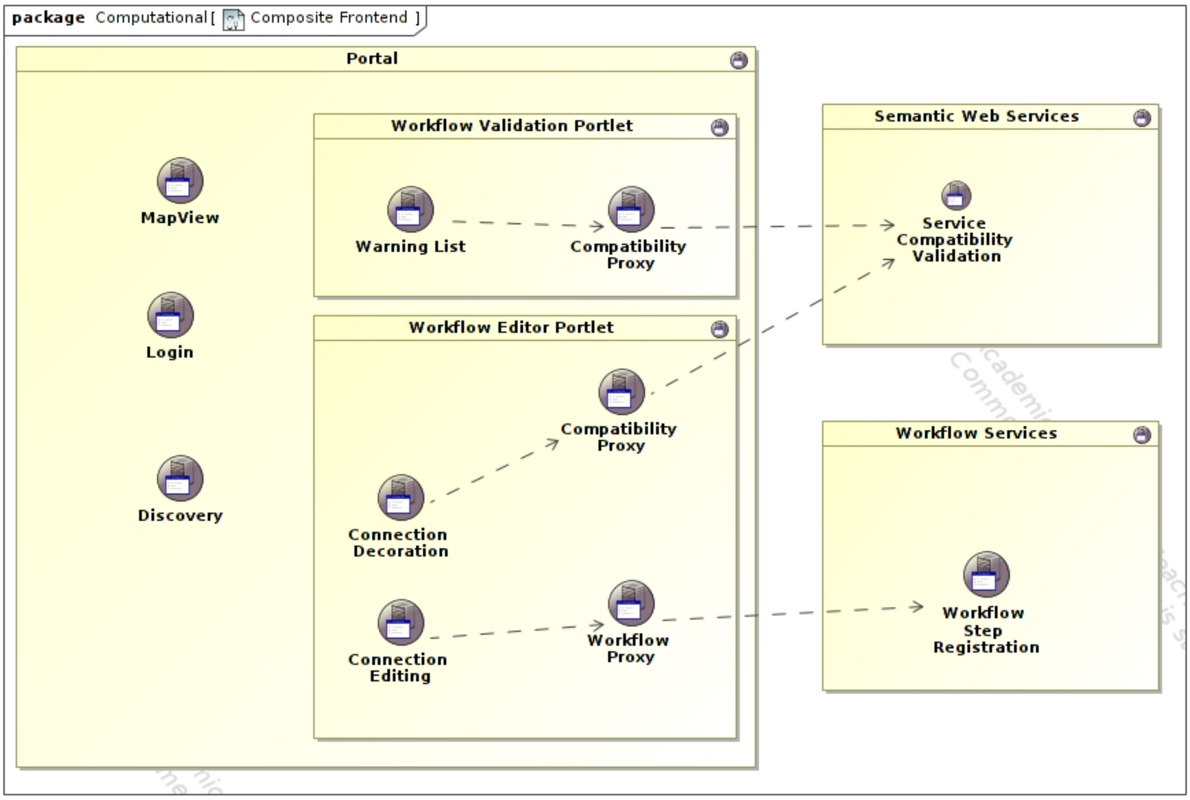


Figure 5-1 Composite Front End pattern as applied to service compatibility validation

The Composite Front End pattern aims to present a cohesive user interface to a number of separate services. It allows services to be designed with the SOA principles of modularity & reuse in mind, while hiding the decoupled nature of the services from the end user, who wishes to see these services combined in a way that meets a need.

In this case, the user is attempting to create a workflow involving a number of separate data and processing services. The user wishes to know if attempting to connect the output of a service to the input of another service makes good semantic sense, e.g. the units are dimensionally compatible.

The Portal provides unified client-side services such as layout and theming and a set of standard core services such as Discovery, MapView and Login. A Workflow Editor Portlet implements the specific composition of services that allows complex workflows to be created and shared.

The UI logic implemented by the Workflow Portlet includes the ability to edit a connection, i.e. attempt to create, delete or modify the pairing of a service output with an input from another service in order to create a link in a service chain. It provides a a feedback mechanism related to service compatibility to the user. A dynamic feedback mechanism of decorating the link between output and input with e.g. a colour to indicate various degrees of compatibility violation. The dynamic feedback is provided within the Workflow Editor Portlet itself.

The UI components must in turn connect to service proxies in order to fulfil their tasks. There is one proxy per service which provides the model for the UI logic. The Compatibility Proxy connects to the Service Compatibility Validation service of the Semantic Web Services, and the Workflow Proxy connects to the Workflow Step Registration service of the Workflow Services.

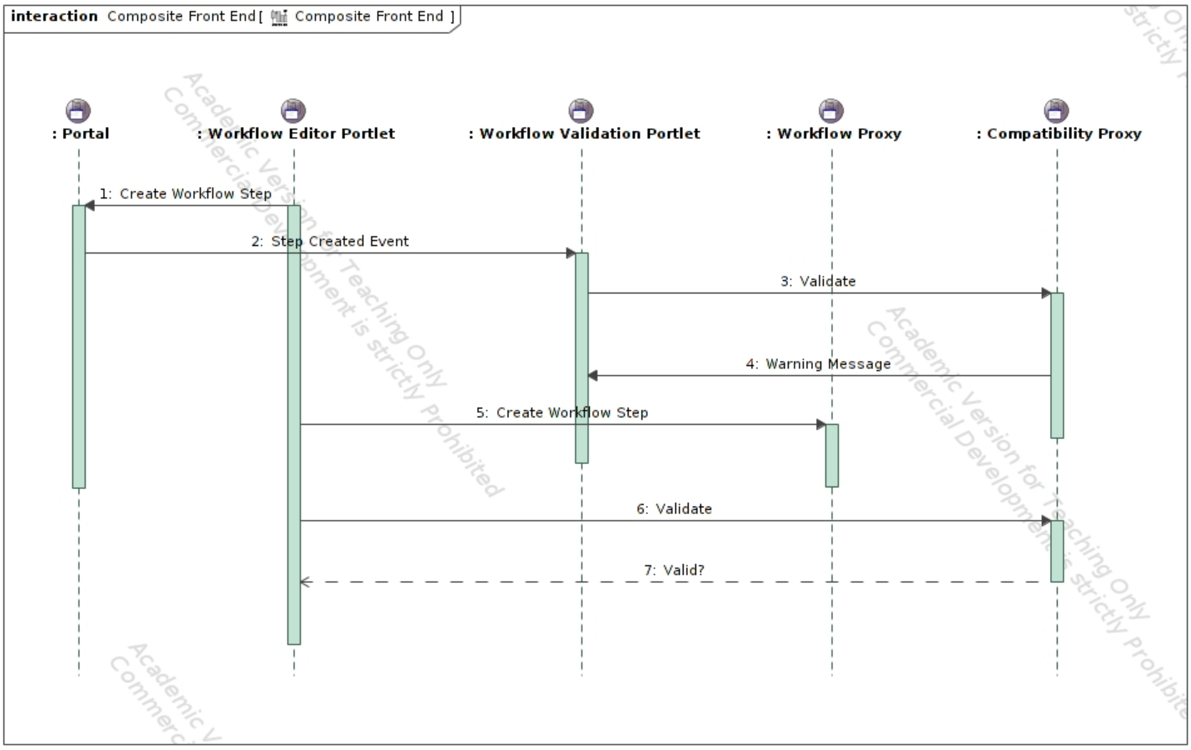


Figure 5-2 Composite Front End pattern interactions

The Workflow Editor Portlet notifies directly its Compatibility Proxy and waits for a response. The response results in direct UI feedback, e.g. display of an icon or decoration on the connection element, or colour coding of the port.

NETMAR uses the Liferay Portal to implement the Composite Front End pattern.

The NETMAR Portal approach promotes the Quality Attributes of Usability and Flexibility. Usability is achieved through reoffering common UI services such as theming and login through a common interface. Flexibility is achieved by allowing the NETMAR core services to be combined in a variety of different ways while maintain both modularity of services and coherence at the application level.

5.1.1 *Validation*

NETMAR's GIS Viewer provides several good examples of this pattern in action. In this case the RichFaces component framework was used to implement a portlet with controls for integrating with external services, e.g. WPS workflow engine and metadata server (Figure 5-3).

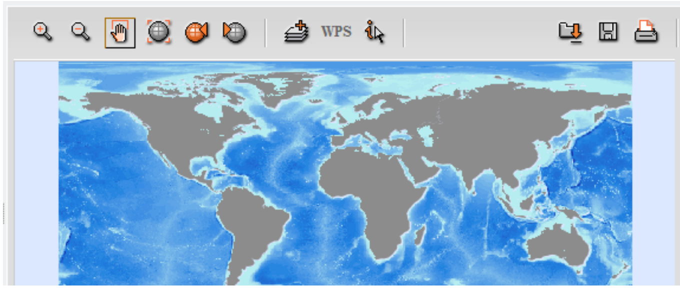


Figure 5-3 - GIS Viewer with AJAX controls

This allows a high degree of interoperability with the other portal components. The user can select the WPS control to execute an existing workflow on input specified by the user, and display the results in the GIS Viewer. From the search component a dataset can be sent to the Viewer for display with a click of a button.

By contrast, the Search component eschewed the portlet pattern and simply embedded a Flex application into a generic canvas element. This resulted in a significant lack of interoperability with the portal. Although a user can display a GIS layer from the Search component, there is no option for passing control in the other direction. If a feature were requested to allow a user to do a semantic search from a keyword in the GIS Viewer, a significant redesign of the search component would be required in order to implement this.

These two contrasting cases – one which follows the pattern and provides a joined-up user experience and one which fails to follow the pattern with consequent negative impact on interoperability – validate the use of this pattern in the context of NETMAR. This does not necessarily validate the choice of technology used to implement the pattern, as there was no control using the same pattern but different technology.

Recommendation: Where multiple services interact, the Composite Front End or Portal pattern should be applied to ensure a coherent user experience.

5.2 Inversion of Communication

A pattern which also works well with this scenario to address some possible performance and usability concerns is Inversion of Communication [RBD11]. In the dynamic editing scenario, the need to call out to an external service while editing may cause the UI to seem slow or unresponsive, depending on network latency, etc. One reason for using the ex post facto validation is that it can be used in a more event driven fashion to remove some of the runtime coupling that could cause performance issues.

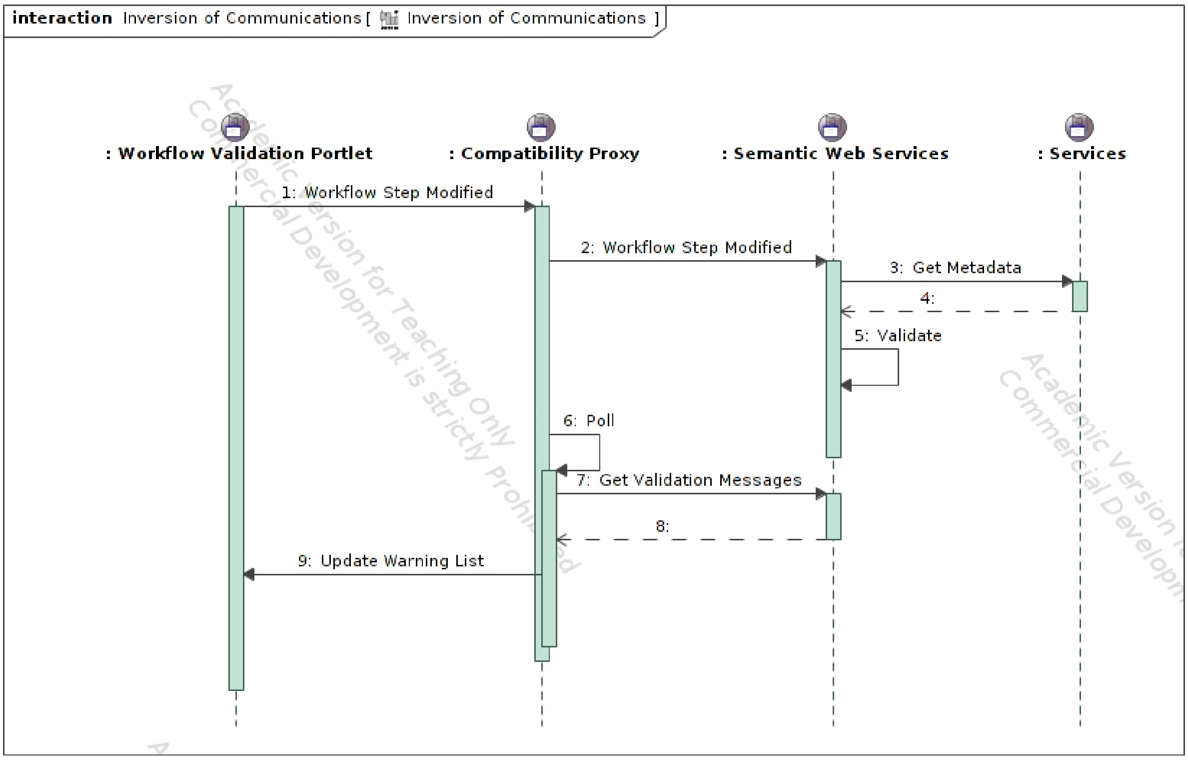


Figure 5-4 Inversion of Communications

Figure 5-4 shows how the backend communications with the Services might work using this pattern. In this case, the Compatibility Proxy pushes notifications of workflow change to the Semantic Web Services. The Semantic Web Services perform all necessary steps to obtain relevant metadata from the services involved in the workflow and validate compatibility, which may involve a considerable delay. As the validation is done, the results of the validation are stored by the Semantic Web services in a queue. The Compatibility Proxy can poll periodically for new validation messages and push them to the Workflow Validation Portlet as they arrive.

This removes some more design and runtime coupling between the Portal and the Semantic Web Services and reduces a possible source of delay on the UI side. There may be a loss in usability, as the feedback from compatibility errors becomes less immediate. If the queue is represented as an RSS or Atom feed, then there is a very simple and direct technology mapping, as the Liferay portal has the ability to publish arbitrary feeds through a built-in widget.

5.2.1 *Validation*

At the time of writing of this report, this functionality was not available for testing.

5.3 *Service Host [RBD11]*

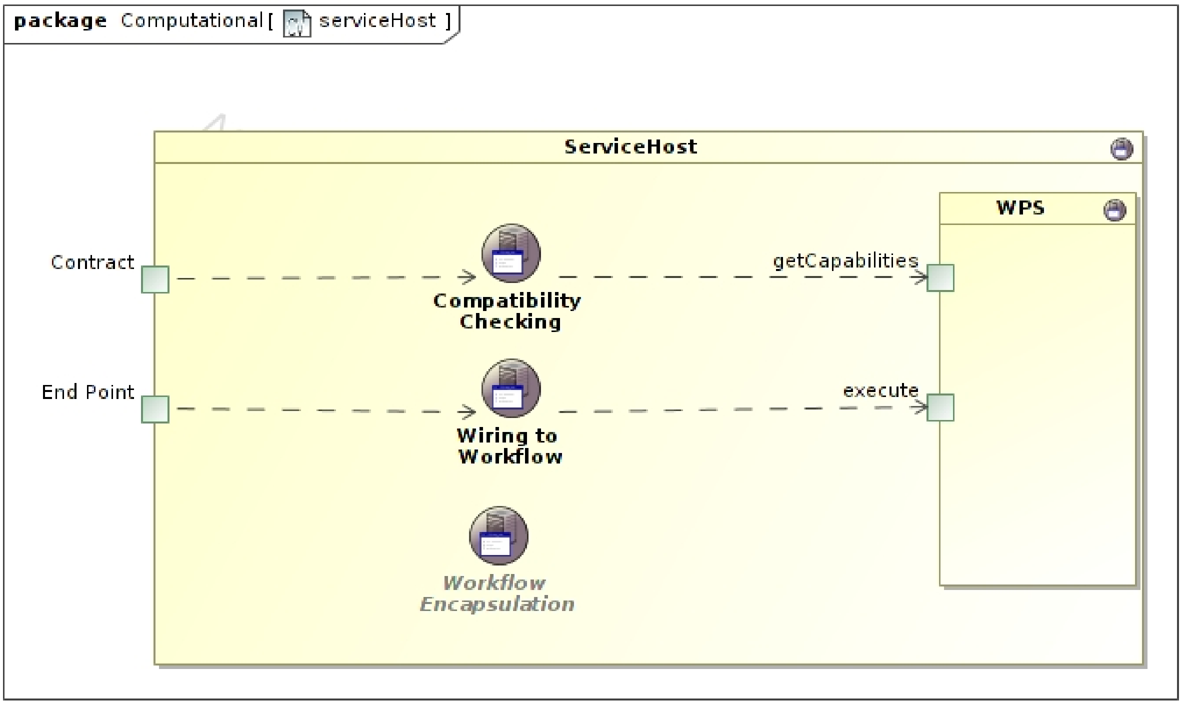


Figure 5-5 Service Host

Each process chain in NETMAR consists of one or more data services along with the processing services which filter and transform these data into a format useful for the end user. Each processing service is modelled on the WPS interface. Along with the basic WPS operations, each processing service must have additional functionality in order to deal with NETMAR specific cases, e.g. semantic checking of dimensions of units for compatibility, configuring endpoints and wiring into workflow, etc.

In order not to have the developer develop this functionality from scratch which would lead to duplication of code and easily avoidable errors, we implement a standard container, based on PyWPS, which takes care of the boilerplate functionality needed to use a processing service within NETMAR.

By doing this we respect the Open Closed Principle, whereby the container is open to extension but closed for modification. For example, a proposed extension of the NETMAR workflow capability is to allow a complete workflow to be encapsulated as a WPS, so that it can be called from within other workflows. The ServiceHost pattern (Figure 5-5) would allow this functionality to be added at a later date, without affecting existing users of the container.

5.3.1 *Validation*

This pattern is closely related to the Enterprise Service Bus pattern [Erl09], in that the ESB is often implemented in terms of a specific service container which abstracts away the details of Service Brokerage, Asynchronous Queuing and Intermediate Routing. In NETMAR, Intermediate Routing was never envisaged as part of the implementation; the absence of any concept of a service inventory meant that point-to-point connections were always going to be the default. A form of asynchronous behaviour was implemented in PyWPS, but it as it was not based in-process but instead relied on creating URLs to permanent files created on the file system, it required a large amount of housekeeping effort on the part of the developer to prevent resource leakage.

In terms of Service Brokerage, PyWPS did provide the ability to convert fairly easily from a WPS getCapabilities document to a WSDL file suitable for loading into an external workflow engine. It also provided the semantic checking of dimensions mentioned above.

Overall, while there were benefits to using PyWPS, and the choice of a uniform container will allow common functionality to be rolled out more easily across the system in future, the validity of this design decision was not fully validated within the lifetime of NETMAR.

Recommendation: When selecting a service container, due attention should be paid to balancing the complexity of deploying the container with the benefits derived from it. A container should provide a significant number of benefits to the developer, be straightforward to deploy, and be suitable for use in the majority of development environments used within the system, otherwise the principle of service autonomy will not be achieved.

5.4 Active Service [RBD11]

The Semantic Web Services have a tight logical coupling to the NERC Vocabulary Server as they depend on the NERC knowledge system to implement their core functionality such as smart search. However, these services are intended to act autonomously, so we do not want to introduce unnecessary runtime coupling. In particular, a user conducting a smart search from the portal should not perceive a delay in getting a response because of a slow network or because the NERC server is temporarily unavailable.

The Active Service pattern (Figure 5-6) is used to address this. The Semantic Web Services will use the AllegroGraph triple store to store SKOS statements retrieved from the NERC vocabulary server. The SWS will periodically retrieve a selection of statements relevant to its needs from the vocab server and internalise them within its own triple store. When the user conducts a smart search which requires the retrieval of facts (either stored or inferred from the stored facts) these will be available without a further network call.

Although this may look like duplication of data, the internal triple store represents added value over the vocabulary server in terms of:

- filtering – the NERC vocabulary server stores many more facts than would be required to meet the Semantic Web Services use cases, therefore only a carefully designed subset of these will be retrieved
- inferencing – in the case of transitive relations, for example, the SWS triple store can use AllegroGraph's inbuilt inferencing ability to infer facts not specifically stated in the knowledge system. More complex inferencing examples can be easily added by adding facts to the SWS store without altering the vocabulary server's contents.

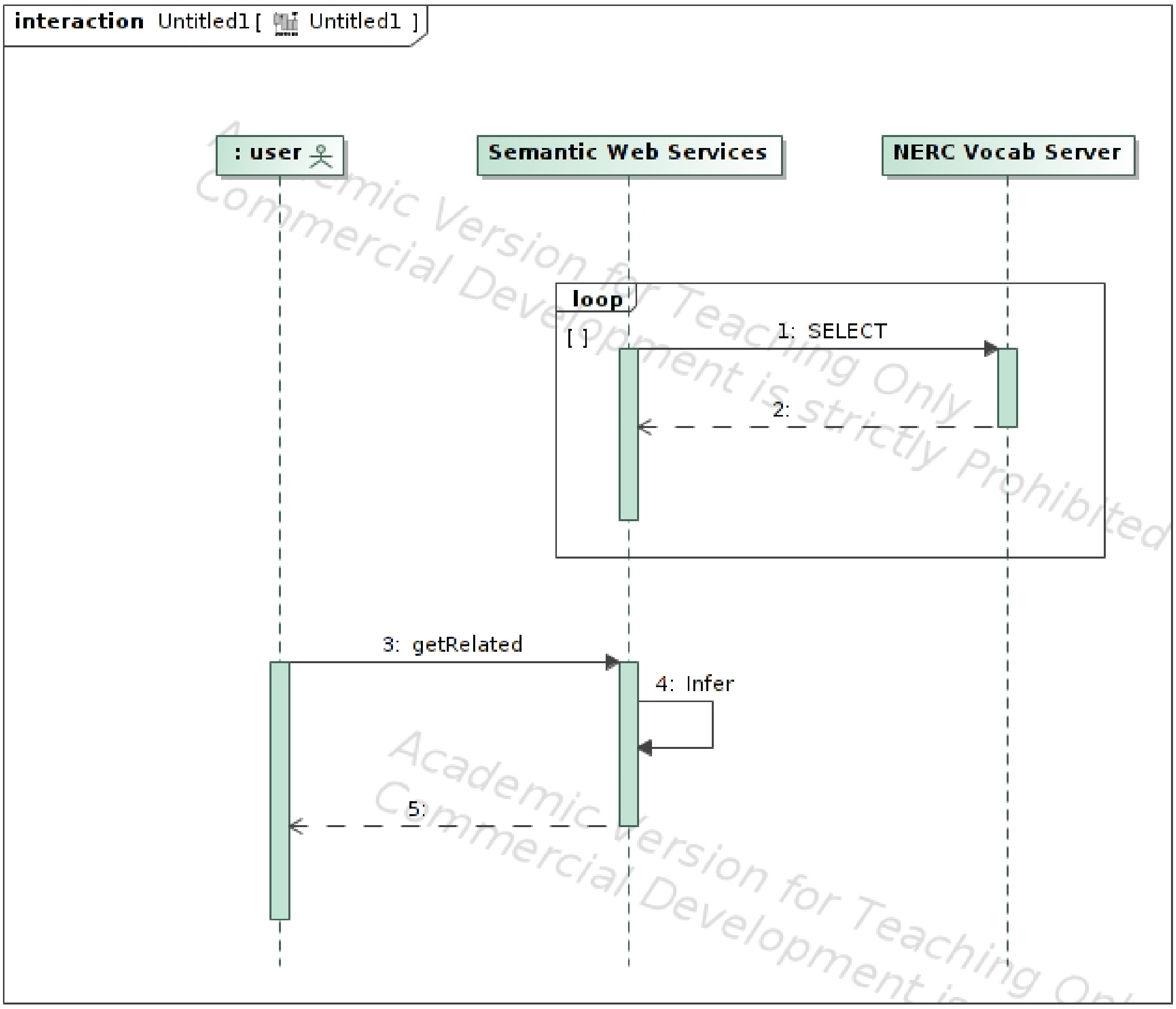


Figure 5-6 Active Service

5.4.1 *Validation*

In terms of implementation, the Semantic Web Services seem in practice to follow a three-tier architecture approach. There is a tight degree of coupling to the NERC Vocabulary Server as the main data layer. There are supplementary connections to metadata repositories; for the most part these seem to be hard-coded within the application. The presentation layer is the Flex application embedded in the Liferay Portal, with SWS acting as the business logic in between. There appears to be a tight coupling between SWS and the Flex application. The one example given of reuse occurred in the context of ICAN, where an almost identical SWS instance was run with an almost identical Flex front-end. This indicates reuse at the code level rather than the service level which was the intention of NETMAR.

Although the service was designed as stateful, using the Jena triple store, not all of the benefits of Active Service seem to have been realised. It is not clear if the latency between NVS and SWS would have been an issue, on the other hand the search functionality seems to exhibit a high degree of sensitivity to availability and location of metadata catalogues, something which could potentially have been addressed with a stateful service.

Recommendation: Technological choices should follow architectural choices, which are specified by the system's stakeholders in the form of quality attributes such as reusability and availability.

6 Conclusions

The aim of this report is to describe the final version of system architecture design for a distributed web-based EIS (Environmental Information System) in line with SOA (service oriented architecture) principles and patterns, and initiatives such as INSPIRE and SISE, and to validate the performance of the architecture against user and system requirements.

System requirements have been identified through interaction with users in four pilot applications. The requirements are to be fulfilled by exposing a core set of capabilities which are within the domain of expertise of the NETMAR partners. These capabilities are described in terms of three main functional areas: common user interface, semantic web services and ontologies, and service chaining editor and workflow engine.

This report outlines the concrete elements of architecture driven by the NETMAR use cases and requirements. It shows specific products, technologies, protocols and decisions being used or intended to be used to implement the first round of user testing and demonstration. These architectural elements were tested during development and testing and conclusions drawn as to their validity. They will be used as the basis for patterns and best practices to be recommended to the wider EIS and software development communities.

This report has described the usage of open source frameworks and tools for deploying CSW, WMS, WFS, WCS, SWE, OPeNDAP, and WPS services. These are:

Protocol	**Tools**
CSW	GeoNetwork
WMS	GeoServer, MapServer, ncWMS, THREDDS (ncWMS)
WFS	GeoServer
WCS	GeoServer, MapServer, THREDDS
SWE	Constellation
OPeNDAP	THREDDS, Dapper
WPS	PyWPS

For the workflow orchestration service, the Taverna open source workflow system for executing process chains was used. This service uses T2Flow XML workflow descriptions.

The Smart Search Service is based upon a bespoke CSW Mediator (CSWM) developed in Java. CSWM is a layer above CSW which enables smart searching of CSW metadata catalogues. As the CSW standard does not directly support smart search, we need this extra layer to support this functionality.

The Semantic Web Service (SWS) provides a high-level interface for retrieving knowledge from the NETMAR ontologies. It supports semantic reasoning for smart discovery of relevant data products and services. It is also used to support semantic validation of service chains in the service chaining editor.

In order that the user can find the services and data with which to construct workflows even where these use a variety of different keywords in different languages, NETMAR uses ontologies based on SKOS made available via ReST and SOAP interfaces.

INSPIRE compliant metadata plays an important role in the NETMAR system. The NETMAR discovery profile is included as an appendix in this report.

NETMAR uses the Liferay portal software to implement a consistent look-and-feel across all products and to ensure consistent user experience while working with multiple independent services. The underlying eventing mechanism allows multiple services to work together in the user's browser to provide the impression of a single coherent product.

The portal includes a Wiki as the mechanism for user support. The Wiki holds all information about products, services and technologies for the pilots.

Portal pilots are comprised of a GIS map viewer component, and optionally a search and discovery client, a service chaining editor and a workflow engine. Together these components provide the functionally requested by the targeted users in the application domains.

The patterns used will be submitted to GEOSS Best Practices Wiki (http://wiki.ieee-earth.org) in a phased approach, subsequent to their having being proven in practice. This will allow for rapid feedback from a wide group of stakeholders.

At the time of writing of this report, insufficient data was available from end users or third party developers of the product to fully validate the main technological choices made. The report has therefore concentrated on a partial validation of choices, centred around established architectural patterns. Where results have indicated that these patterns worked, we have recommended their use. Where results appear lacking or inconclusive, we have recommended a course of action likely, based on existing software practice, to lead to more conclusive results. Our core recommendations are:

- Focus on patterns, principles and practices rather than formal frameworks.
- Apply standard software engineering practices to ensure an adequate level of quality.
- Use documented patterns in order to learn from the mistakes of others.
- Ensure a joined-up user experience by using the Composite Front End pattern.
- Choose service containers wisely, balancing functionality against added complexity.
- Do not allow technology to drive the project.

7 References

[ASMG09] Atenea System Modelling Group, 2009. UML4ODP Plugin User Guide Version 0.9.

[ACRI09] ACRI, 2009. D8100.3- Coastal Water and Air Quality Thematic Pilot Specification Document. GENESIS Deliverable D8100.3, issue 1.0, 15/07/2009. Online at http://genesis-fp7.eu/publications (accessed 6 April 2010).

[BD10] Brugge, Bernd, and Allen H. Dutoit, 2010. Object-Oriented Software Engineering Using UML, Patterns and Java. Third edition. Pearson Education, Inc., 2010. ISBN 13: 978-0-13-815221-5.

[BN11] Bigagli, Lorenzo and Stefano Nativi, 2011. NetCDF Uncertainty Conventions (NetCDF-U) 1.0. OGC Discussion Paper. OGC 11-163. Online at http://www.unidata.ucar.edu/mailing_lists/archives/galeon/2011/pdfhhGKdi3Uu6.pdf (access on 10 February 2012).

[Bus96] F. Buschmann, R. Meunier, H. Rohnert, P. Sommerlad, and M. Stal, Pattern-Oriented Software Architecture - A System of Patterns. Wiley and Sons, 1996

[Crn10] Crnkovic, G. Constructive Research and Info-Computational Knowledge Generation, Model-Based Reasoning in Science and Technology, pp. 359-380, 2010. [Online].

[CSW202] OpenGIS Catalogue Service Specification, Version 2.0.2, OGC 07-006r1. Online at http://portal.opengeospatial.org/files/?artifact_id=20555 (accessed on 22 April 2010).

[DAP+10] Dunne, D. et al. 2010. Review of projects, initiatives & technologies addressing system architectures for distributed Environmental Information Systems (EIS). NETMAR (Open service network for marine environmental data) Deliverable D2.1. European Commission Information Society and Media Directorate-General Grant Agreement Number 249024.

[DLR09] DLR, 2009. D2100.3- Air Quality Thematic Pilot Specification Document. GENESIS Deliverable D2100.3, issue 1.0, 15/07/2009.
Online at http://genesis-fp7.eu/publications (accessed 6 April 2010).

[DOD03] US Dept of Defence, 2003. DoD Architecture, Volume 1: Definitions and Guidelines, Version1

[EGF+10] B. Ellinger, N. Grumman, L. Fehskens, C. Harding, E. Harrington, F. McCabe, and B. Miner, "Navigating the SOA Open Standards Landscape Around Architecture." [Online]. Available: http://www.adobe.com/content/dam/Adobe/en/devnet/livecycle/pdfs/soa_standards.pdf

[Erl09] Thomas Erl, "SOA Design Patterns", Prentice Hall, 2009

[GC09a] GIGAS Consortium, 2009a. GIGAS Technology Watch Report Summary Technical Note. Deliverable reference D2.2b, revision 103, 10.12-2009. Online at http://www.thegigasforum.eu/cgi-bin/download.pl?f=354.pdf (accessed 26 March 2010).

[GC09b] GIGAS Consortium, 2009b. GIGAS Comparative Analysis Technical Note. Deliverable reference D2.3b, revision 200, 14-12-2009. Online at http://www.thegigasforum.eu/cgi-bin/download.pl?f=359.pdf (accessed 26 March 2010).

[GC09c] GIGAS Consortium, 2009c. GIGAS Cross-Initiative Scenarios. Deliverable Reference D2.2b, revision 104. Online at http://www.thegigasforum.eu/cgi-bin/download.pl?f=350.pdf (accessed 26 March 2010)

[GC09d] GIGAS Consortium, 2009d. GIGAS Technology Watch Report Architecture Technical Note. Deliverable reference D2.2b, version 103, 15-11-2009. Online at http://www.thegigasforum.eu/cgi-bin/download.pl?f=345.pdf (accessed 26 March 2010).

[GC09e] GIGAS Consortium, 2009e. GIGAS Comparative Analysis Annex. Deliverable reference D2.3b (Annex), version 200, 14-12-2009. Online at http://www.thegigasforum.eu/cgi-bin/download.pl?f=360.pdf (accessed 30 March 2010).

[GEN11] GENESIS Consortium, 2011. The GENESIS Solution. Generic European Sustainable Information Space for Environment. [Online] Available: http://www.genesis-fp7.eu/images/publications/leaflets/brochure.pdf

[GSDI09] Wilson, M. (Ed), 2009, Chapter Three: Metadata -- Describing geospatial data, Spatial Data Infrastructure Cookbook, Global Spatial Data Infrastrucutre (GSDI), [Online] Available: http://memberservices.gsdi.org/files/?artifact_id=655

[INS07a] DT Metadata, Draft Implementing Rules for Metadata, version 2. Online at http://inspire.jrc.ec.europa.eu/reports/ImplementingRules/draftINSPIREMetadataIRv2_20070202.pdf

[INS07b] Data Specifications, Network Services and Metadata Drafting Team members, 2007, INSPIRE Technical Architecture – Overview, Online at http://inspire.jrc.ec.europa.eu/reports/ImplementingRules/network/INSPIRETechnicalArchitectureOverview_v1.2.pdf (06 January 2012)

[Hreb09] Hřebíčeka, J., Pillmannb, W. 2009, Shared Environmental Information System and Single Information Space in Europe for the Environment: Antipodes or Associates?, Opportunities of SEIS and SISE: Integrating Environmental Knowledge in Europe, European conference of the Czech Presidency of the Council of the EU TOWARDS eENVIRONMENT. Online at http://www.epractice.eu/files/SEIS%20and%20SISE%20for%20the%20Environment_Antipodes%20or%20Associates.pdf (accessed 19 April 2010).

[ISO09] Use of UML for ODP system specifications. ITU-T Recommendation X.906 | ISO/IEC 19793

[ISO10] ITU-T X.903 | ISO/IEC 10746-3 Information Technology – Open Distributed Processing – Reference Model – Architecture

[DJW11] de Jesus, Jorge and Peter Walker, 2011. D7.7 WPS Cookbook. NETMAR (Open service network for marine environmental data) Deliverable D7.7. European Commission Information Society and Media Directorate-General Grant Agreement Number 249024.

[Kru95] Kruchten, P.B. 1995. The 4+1 View Model of Architecture, *IEEE Software* 12 (6) p. 42-50

[Lin10] Linington, Peter. The Reference Model of Open Distributed Processing : Foundations , Experiences and Applications. Computer Standards & Interfaces

[LLC+11] Lowry, R., et al. 2011. Strategy for ontology and tool development to fulfil NETMAR requirements. NETMAR (Open service network for marine environmental data) Deliverable D3.4. European Commission Information Society and Media Directorate-General Grant Agreement Number 249024.

[Mar99] Martin, R, 1999. Iterative and Incremental Development (IID)

[MB09] Miles, A., and Bechhofer, S.: SKOS Simple Knowledge Organization System: Reference. W3C Recommendation. 18 August 2009. Available online at: http://www.w3.org/TR/2009/REC-skos-reference-20090818/.

[OAS06] OASIS, 2006. Reference Model for Service Oriented Architecture 1.0

[OGC07] Open Geospatial Consortium, 2007. Reference Model for the ORCHESTRA Architecture (RM-OA) V2 (Rev2.1)

[OMG05] Object Management Group. Unified Modeling Language: Superstructure

[OMG08] Object Management Group, 2008. Business Process Modeling Notation, V1.1

[OMG09] Object Management Group 2010. Service oriented architecture Modeling Language (SoaML)-Specification for the UML Profile and Metamodel for Services (UPMS)

[OMG09a] Object Management Group, 2009. Ontology Definition Metamodel. V1.0

[PHD+12] Patterson et. Al., 2012. D2.2.2 NETMAR system architecture design – Final version. NETMAR (Open service network for marine environmental data) Deliverable D2.2.2. European Commission Information Society and Media Directorate-General Grant Agreement Number 249024.

[PTS+11] Pladsen, Jan Ivar, et. al, 2011. D1.1 Use case specification and schedule for testing. NETMAR (Open service network for marine environmental data) Deliverable D1.1. European Commission Information Society and Media Directorate-General Grant Agreement Number 249024.

[RBD11] Rotem-Gal-Oz, Bruno, Dahan. SOA Patterns. Manning 2011.

[Shaw01] Shaw, M. 2001. The Coming-of-Age of Software Architecture Research. Proceedings of ICSE-2001, pp. 657-664. Los Alamitos, CA: IEEE Computer Society Press.

[SEISa] Shared Environmental Information System (SEIS). Online at http://ec.europa.eu/environment/seis/index.htm (accessed 06 January 2012).

[SEISb] Shared Environmental Information System (SEIS). Online at http://www.eea.europa.eu/about-us/what/shared-environmental-information-system (accessed 06 January 2012).

[SISE2] Towards a Single Information Space for the Environment in Europe, Experts Consultation Workshop. Online at http://cordis.europa.eu/fp7/ict/sustainable-growth/workshops/ws-20080215_en.html (accessed 19 April 2010).

[SGH09] SGH, 2009. D3100.3- Fresh Water Quality Thematic Pilot Specification Document. GENESIS Deliverable D3100.3, issue 1.0, 15/07/2009. Online at http://genesis-fp7.eu/publications (accessed 6 April 2010).

[SOS06] OGC Sensor Observation Service Implementation Standard Version 1.0, OGC OGC 06-009r6. Online at http://www.opengeospatial.org/standards/sos (accessed 6 January 2012)

[T+11] Torget et al, Evaluation of NETMAR system applied to use cases – First version. NETMAR (Open service network for marine environmental data) Deliverable D1.3.1. European Commission Information Society and Media Directorate-General Grant Agreement Number 249024.

[TAN10] W. Tan, P. Missier, I. Foster, R. Madduri, D. De Roure, and C. Goble, "A comparison of using Taverna and BPEL in building scientific workflows: the case of caGrid," Concurrency and Computation: Practice and Experience, vol. 22, iss. 9, pp. 1098-1117, 2010.

[TOGAF] TOGAF, The Open Group Architecture Framework, version 8.1. Online at http://www.opengroup.org/architecture/togaf8-doc/arch/toc.html (accessed 9 November 2010).

[UML23] Object Management Group, 2010. UML version 2.3. Online at http://www.omg.org/spec/UML/2.3/ (accessed 12 June 2011)

[W3C10] RIF Use Cases and Requirements (http://www.w3.org/2005/rules/wiki/UCR)

[WCS11] OGC Web Coverage Service (WCS) Implementation Specification, version 1.1.2, OGC 07-067r5, Online at http://portal.opengeospatial.org/files/?artifact_id=27297 (accessed 22 April 2010)

[WFS04] OGC Web Feature Service Implementation Specification, version 1.1.0, OGC 04-094. Online at http://www.opengeospatial.org/standards/wfs (accessed 5 January 2012)

[WMC05] OGC Web Map Context Documents Implementation Specification, version 1.1.0, OGC 05-005. Online at http://www.opengeospatial.org/standards/wmc (accessed 5 January 2012)

[WMS13] OGC Web Map Service (WMS) Implementation Specification, version 1.3.0, OGC 06-042, Online at http://portal.opengeospatial.org/files/?artifact_id=14416 (accessed 21 April 2010)

[WK03] Wood, W.A., Kleb, W.L., Exploring XP for scientific research, Software, IEEE , vol.20, no.3, pp. 30- 36, May-June 2003 doi: 10.1109/MS.2003.1196317

[WPS10] OGC Web Processing Service, version 1.0.0, OGC 05-007r7. Online at http://portal.opengeospatial.org/files/?artifact_id=24151 (accessed 26 April 2010)

[Zac87] Zachman, J. A., 1987. A framework for information systems architecture, IBM Systems Journal, 26 (3)

Appendices

Appendix A. NETMAR Discovery Metadata Profile

This appendix contains the NETMAR metadata profile. Its scope covers 3 metadata hierarchy levels:

1. Dataset
2. Dataset series (dataset collections)
3. Service metadata

A dataset is defined by ISO 19115 as an: "identifiable collection of data". A dataset series is defined by ISO 19115 as a: "collection of datasets sharing the same product specification". Metadata for which no hierarchy is listed are interpreted to be "dataset" metadata, by default. The hierarchical relationship between dataset and dataset series is illustrated in *Figure* A-*1*. Feature and attribute metadata are not required by the NETMAR system. Instead the main requirements are the documentation of feature and/or attribute parameter and unit types within the dataset or dataset series metadata.

Service metadata is defined by ISO 19119 as: "a service metadata record describes a service instance, including a description of the services operations and an 'address' to access the specific service instance".

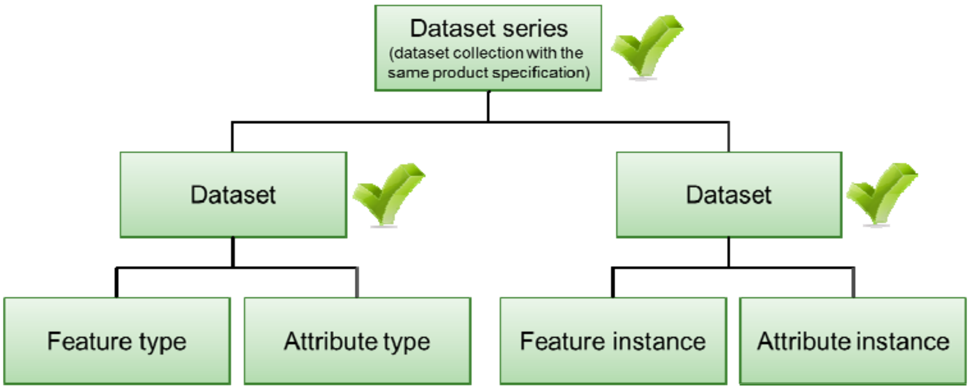


Figure A-*1 Metadata hierarchy (reference: ISO 19115:2003)*

The development of the NETMAR metadata profile model is based on analysis of the following standards, legislation, and guidance documents:

1. ISO 19115:2003, Geographic information - Metadata
2. ISO 19115:2003/Cor. 1:2006, Geographic information - Metadata - Technical Corrigendum 1
3. ISO 19115-2, Geographic information - Metadata - Part 2: Metadata for imagery and gridded data
4. ISO/CD 19115-1, Geographic information - Metadata - Part 1: Fundamentals (Draft, 2011-01-24)
5. ISO 19119:2005, Geographic information - Services
6. ISO 19119:2005/Amd 1:2008, Extensions of the service metadata model
7. ISO 19139:2007, Geographic information - Metadata – XML Schema Implementation
8. ISO 19103:2005, Geographic information - Conceptual schema language

9. OpenGIS Catalogue Services Specification 2.0.2 - ISO Metadata Application Profile, Version 1.0.0, OGC 07-045, 2007
10. Commission Regulation (EC) No 1205/2008 of 3 December 2008 implementing Directive 2007/2/EC of the European Parliament and of the Council as regards metadata
11. Corrigendum to INSPIRE Metadata Regulation published in the Official Journal of the European Union, L 328, page 83
12. INSPIRE Metadata Implementing Rules: Technical Guidelines based on EN ISO 19115 and EN ISO 19119 (Version 1.2)
13. INSPIRE Generic Conceptual Model (D 2.5, Version 3.3)

Also, because the FP7 NETMAR consortium is inherently based on the completed FP6 InterRisk project, metadata profile analysis was also conduced on the following document:

14. D3-2B InterRisk Discovery Metadata Profile

Finally, the following two UK metadata profile projects, compliant with INSPIRE, also proved to be useful resources to assist NETMAR metadata profile work:

15. UK Gemini Encoding Guidance, version 1.0, September 2010 (http://location.defra.gov.uk/wp-content/uploads/2010/09/UK-GEMINI-Encoding-Guidance-20100930-v1-01.pdf)

16. Guidance notes for the production of discovery metadata for the Marine Environmental Data and Information Network (MEDIN), version 2.3.4 (http://www.oceannet.org/marine_data_standards/medin_approved_standards/documents/medin_schema_doc_2_3_4_30nov2010.pdf)

The following sections list the ISO 19115/19119 metadata profile for NETMAR. Informal dictionary tables are used to document, analyse, and comment on the metadata model. The numbering system for the dictionary tables is the aligned to the exact numbers for corresponding metadata entities / elements in ISO 19115 and 19119.

A 1. MD_Metadata

ISO	NETMAR Metadata Profile	NETMAR MD scope	ISO 19115/19119 analysis	InterRisk analysis	INSPIRE analysis	Comments
1	MD_Metadata		Lines 2-22			
2	+ fileIdentifier [1] : CharacterString	dataset & series, service	[0..1]	[1]	-	ISO Core element. Required by CSW2 ISO AP. Future ISO 19115-2 Draft comment: Moved to MD_Metadata/metadataIdentifier
3	+ language [1] : LanguageCode	dataset & series, service	[0..1] C / not defined by encoding?	[1]	Metadata language [1]	LanguageCode dataType is from ISO 19139. ISO Core element. INSPIRE SC1: Metadata language is mandatory. INSPIRE Guidelines: language codelist based on alpha-3 codes of ISO 639-2. Use only 3 letter codes.
4	+ characterSet [1] : MD_CharacterSetCode	dataset & series, service	[0..1] C / ISO/IEC 10646-1 not used and not defined by encoding? (utf-8)	[1]	-	ISO Core element. INSPIRE note: Conditional for most INSPIRE data specifications concerning dataset and dataset-series.
5	+ parentIdentifier [0..1] : CharacterString (C)	dataset	[0..*] C / If there is an upper hierarchy level	[1]	-	NETMAR Guidelines: C / If there is an upper hierarchy level. Future ISO 19115-2 Draft comment: Moved to MD_Metadata/parentMetadataIdentifier

6	+ hierarchyLevel [1] : MD_ScopeCode	dataset & series, service	[0..*] C / hierarchyLevel is not equal to "dataset"?	[1]	Resource Type [1]	INSPIRE SC2: Hierarchy Level is mandatory. INSPIRE SC3: INSPIRE only considers the first instance. INSPIRE SC4: If the value is not service, dataset or series, then metadata is out of scope of the INSPIRE Directive. Future ISO 19115-2 Draft comment: Moved to MD_Metadata/metadataScope
8	+ contact [1..*] : CI_ResponsibleParty	dataset & series, service	[1..*]	[1..*]	Metadata point of contact [1..*]	Definition: party responsible for the metadata information ISO Core element. **Refer to Dictionary Table: A 2 - CI_ResponsibleParty**
9	+ dateStamp [1] : Date | DateTime	dataset & series, service	[1]	[1]	Metadata date [1]	Date and DataTime dataTypes from ISO 19139. ISO Core element. Future ISO 19115-2 Draft comment: Renamed to dateInfo, and cardinality changed to [1..*]
10	+ metadataStandardName [1] : CharacterString	dataset & series, service	[0..1]	[1]	-	ISO Core element Future ISO 19115-2 Draft comment: Renamed to metadataStandard, and uses Class CI_Citation
11	+ metadataStandardVersion [1] : CharacterString	dataset & series, service	[0..1]	[1]	-	ISO Core element Future ISO 19115-2 Draft comment: Moved to metadataStandard, and uses Class CI_Citation
12	+ spatialRepresentationInfo [0..*] : MD_SpatialRepresentation	dataset & series, service	[0..*]	[0..*]	-	**Refer to Dictionary Table: A 3 - MD_SpatialRepresentation**

13	+ referenceSystemInfo [1..*] : MD_ReferenceSystem	dataset & series, service	[0..*]	[0..*]	Coordinate Reference System (C, see INSPIRE note 1) And Temporal Reference System (C, see INSPIRE note 2)	ISO Core element INSPIRE note: 1) Mandatory for most INSPIRE data specifications concerning dataset and dataset-series. 2) Conditional for most INSPIRE data specifications concerning dataset and dataset-series, if time not Gregorian or Coordinated UT (note: Temporal Reference System not explicitly supported by ISO 19115:2002) **Refer to Dictionary Table:** **A 4 - MD_ReferenceSystem**
15	+ identificationInfo [1] : MD_Identification	dataset & series, service	[1..*]	[1..*]	[1]	INSPIRE SC5: When there are many instances only the first one concerns the INSPIRE resource. **Refer to Dictionary Table:** **A 5 - MD_Identification**
17	+ distributionInfo [0..1] : MD_Distribution (C)	dataset & series, service	[0..1]	[0..1]	[0..1]	INSPIRE Guidelines: For dataset and dataset series, mandatory if a URL is available to obtain more information on the resources and/or access related services. For services, mandatory if linkage to the service is available. NETMAR Guidelines: Use same condition rules as INSPIRE. Future ISO 19115-2 Draft comment: Cardinality changed to [0..*] **Refer to Dictionary Table:** **A 10 - MD_Distribution**

18	+ dataQualityInfo [0..*] : DQ_DataQuality	dataset & series, service	[0..*]	[0..]	[0..]	INSPIRE SC6: INSPIRE only considers the instance applicable to the whole resource. **Refer to Dictionary Table: A 12 - DQ_DataQuality**
20	+ metadataConstraints [0..*] : MD_Constraints	dataset & series, service	[0..*]	[0..*]	-	**Refer to A 14 - MD_Constraints (for metadataConstraints)**

A 2. CI_ResponsibleParty

#	NETMAR Metadata Profile	NETMAR MD scope	ISO 19115/19119 analysis	InterRisk analysis	INSPIRE analysis	Comments
374	CI_ResponsibleParty		Lines 375-379			Future ISO 19115-2 Draft comments: Class renamed CI_Responsibility. Class restructured.
375	+ individualName [0..1] : CharacterString	dataset & series, service	[0..1] C / organisationName and positionName not documented?	-	-	
376	+ organisationName [1] : CharacterString	dataset & series, service	[0..1] C / individualName and positionName not documented?	-	[1] for SC14 and SC15	INSPIRE SC14: MD_Metadata.identificationInfo[1]. MD_DataIdentification.pointOfContact[1]. CI_ResponsibleParty.organisationName is mandatory. INSPIRE SC14: MD_Metadata.identificationInfo[1]. SV_ServiceIdentification.pointOfContact[1]. CI_ResponsibleParty.organisationName is mandatory. INSPIRE SC15: MD_Metadata.contact[1]. CI_ResponsibleParty.organisationName is mandatory.
377	+ positionName [0..1] : CharacterString	dataset & series, service	[0..1] C / individualName and organisationName not documented?	-	-	
378	+ contactInfo [1] : CI_Contact	dataset & series, service	[0..1]	-	[1] for SC14 and SC15	
>388	+ phone [0..1] : CI_Telephone	dataset & series, service	[0..1]	-	-	
>>408	+ voice [0..*] : CharacterString	dataset & series, service	[0..*]	-	-	

>>409	+ facsimile [0..*] : CharacterString	dataset & series, service	[0..*]	-	-	
>389	+ address [1] : CI_Address	dataset & series, service	[0..1]	-	[1] for SC14 and SC15	
>>381	+ deliveryPoint [0..*] : CharacterString	dataset & series, service	[0..*]	-	-	
>>382	+ city [0..1] : CharacterString	dataset & series, service	[0..1]	-	-	
>>383	+ administrativeArea [0..1] : CharacterString	dataset & series, service	[0..1]	-	-	
>>384	+ postalCode [0..1] : CharacterString	dataset & series, service	[0..1]	-	-	
>>385	+ country [0..1] : CharacterString	dataset & series, service	[0..1]	-	-	ISO 3166-3
>>386	+ electronicMailAddress [1..*] : CharacterString	dataset & series, service	[0..*]	-	[1..*] for SC14 and SC15	INSPIRE SC14: MD_Metadata.identificationInfo[1]. MD_DataIdentification.pointOfContact[1]. CI_ResponsibleParty.contactInfo. CI_Contact.address.CI_Address. electronicMailAddress is mandatory. INSPIRE SC14: MD_Metadata.identificationInfo[1]. SV_ServiceIdentification.pointOfContact[1]. CI_ResponsibleParty.contactInfo. CI_Contact.address.CI_Address. electronicMailAddress is mandatory. INSPIRE SC15: MD_Metadata.contact[1]. CI_ResponsibleParty.contactInfo. CI_Contact.address.CI_Address. electronicMailAddress is mandatory.
>389	+ onlineResource [0..1] : CI_OnlineResource	dataset & series, service	[0..1]	-	-	
>>397	+ linkage [1] : URL	dataset & series, service	[1]	-	-	

379	+ role [1] : CI_RoleCode	dataset & series, service	[1]	-	[1] for SC16	INSPIRE SC16: The value of MD_Metadata.contact[1]. CI_ResponsibleParty.role.CI_RoleCode shall be “pointOfContact”.

A 3. MD_SpatialRepresentation

#	NETMAR Metadata Profile	NETMAR MD scope	ISO 19115/19119 analysis	InterRisk analysis	INSPIRE analysis	Comments
156	MD_SpatialRepresentation					
157	MD_GridSpatialRepresentation		Lines 158-161			Support for Grid
158	+ numberOfDimensions [1] : Integer	dataset & series, service	[1]	-	-	
159	+ axisDimensionProperties [1] : Sequence<MD_Dimension>	dataset & series, service	[1]	-	-	
>180	+ dimensionName [1] : MD_Dimension	dataset & series, service	[1]	-	-	
>181	+ dimensionSize [1] : Integer	dataset & series, service	[1]	-	-	
>182	+ resolution [1] : Integer	dataset & series, service	[1]	-	-	
160	+ cellGeometry [1] : MD_CellGeometryCode	dataset & series, service	[1]	-	-	
161	+ transformationParameterAvailability[1] : Boolean	dataset & series, service	[1]	-	-	
176	MD_VectorSpatialRepresentation		Lines 177-178			Support for Vector
177	+ topologyLevel [0..1] : MD_TopologyLevelCode	dataset & series, service	[0..1]	-	-	
178	+ geometricObjects [1] : MD_GeometricObjects	dataset & series, service	[0..1]	-	-	
>184	+ geometricObjectType [1] : MD_GeometricObjectTypeCode	dataset & series, service	[1]	-	-	

A 4. MD_ReferenceSystem

#	NETMAR Metadata Profile	NETMAR MD scope	ISO 19115/19119 analysis	InterRisk analysis	INSPIRE analysis	Comments
186	MD_ReferenceSystem					
187	+ referenceSystemIdentifier [0..1] : RS_Identifier	dataset & series, service	[0..1] Refer to SC_CRS in ISO 19111 when coordinate reference system information is not given through reference system identifier	-	-	
>207	+ code [1] : CharacterString	dataset & series, service	[1]	-	-	NETMAR Guidelines: Use EPSG codes for CRS (unless using an INSPIRE RS registry).
>208.1	+ codeSpace [0..1] : CharacterString	dataset & series, service	[0..1]	-	-	

A 5. MD_Identification (including MD_DataIdentification and SV_ServiceIdentification)

#	NETMAR Metadata Profile	NETMAR MD scope	ISO 19115/19119 analysis	InterRisk analysis	INSPIRE analysis	Comments
23	MD_Identification					
24	+ citation [1] : CI_Citation	dataset & series, service	[1]	-	[1]	**Refer to Dictionary Table:** **A 6 - CI_Citation (for MD_Identification.citation)**
25	+ abstract [1] : CharacterString	dataset & series, service	[1]	-	Resource abstract [1]	ISO Core element.
26	+ purpose [0..1] : CharacterString	dataset & series, service	[0..1]	-	-	
28	+ status [0..*] : MD_ProgressCode	dataset & series, service	[0..*]	-	-	
29	+ pointOfContact [1..*] : CI_ResponsibleParty	dataset & series, service	[0..*]	-	Responsible party [1]	Definition: identification of, and means of communication with, person(s) and organization(s) associated with the resource(s) ISO Core element. **Refer to Dictionary Table:** **A 2 - CI_ResponsibleParty**
30	+ resourceMaintenance [0..*] : MD_MaintenanceInformation	dataset & series, service	[0..*]	-	-	
31	+ graphicOverview [0..*] : MD_BrowseGraphic	dataset & series, service	[0..*]	-	-	
>49	+ fileName [1] : MD_BrowseGraphic	dataset & series, service	[1]	-	-	
>50	+ fileDescription [0..1] : CharacterString	dataset & series, service	[0..1]	-	-	
>51	+ fileType [0..1] : CharacterString	dataset & series, service	[0..1]	-	-	
32	+ resourceFormat [0..*] : MD_Format	dataset & series, service	[0..*]	-	-	
>285	+ name [1] : CharacterString	dataset & series, service	[1]	-	-	INSPIRE example (for distributionFormat): e.g. “Protected sites GML application schema”.
>286	+ version [1] : CharacterString	dataset & series, service	[1]	-	-	INSPIRE example (for distributionFormat): e.g. “version 3.1; GML, version 3.2.1”.

>288	+ specification [1] : CharacterString	dataset & series, service	[0..1]	-	-	INSPIRE example (for distributionFormat): e.g. “D2.8.I.9 Data Specification on Protected sites – Guidelines”.
>289	+ fileDecompressionTechnique [0..1] : CharacterString	dataset & series, service	[0..1]	-	-	
33	+ descriptiveKeywords [1..*] : MD_Keywords	dataset & series, service	[0..*]	-	Keyword (M)	INSPIRE SC17: For datasets and series at least one keyword of GEMET thesaurus shall be documented. INSPIRE SC18: For services at least one keyword of Part D.4 of Commission Regulation (EC) No. 1205/2008 shall be documented. **Refer to Dictionary Table:** **A 7 - MD_Keywords**
35	+ resourceConstraints [1..*] : MD_Constraints	dataset & series, service	[0..*]	-	Constraints related to access and use (M)	INSPIRE SC12: There shall be at least one instance of resourceConstraints. INSPIRE Guidelines: Each instance of MD_Constraints expresses: • 0 or 1 condition applying to access and use • 0 or more limitations on public access • Or, both one or more limitations on public access and a condition applying to access and use. INSPIRE Guidelines: In any case, there shall be at least one ISO 19115 metadata element representing a limitation on public access and one ISO 19115 metadata element representing a condition applying to access and use as part of the different instances of MD_Constraints and its subclasses. **Refer to Dictionary Table:** **A 13 - MD_Constraints (for MD_Identification.resourceConstraints)**

36	MD_DataIdentification		Lines 37-46 and 24-35.1			
37	+ spatialRepresentationType [0..*] : MD_SpatialRepresentationTypeCode	dataset & series	[0..*]	-	-	ISO Core element
38	+ spatialResolution [0..*] : MD_Resolution (C)	dataset & series	[0..*]	-	[0..*] Spatial Resolution	INSPIRE Guidelines: Mandatory if an equivalent scale or a resolution distance can be specified. NETMAR Guidelines: Use same condition rules as INSPIRE. **Refer to Dictionary Table:** **A 8 - MD_Resolution**
39	+ language [1..*] : CharacterString	dataset & series	[1..*]	-	Resource language [0..*]	INSPIRE Guidelines: Mandatory if the resource includes textual information.
40	+ characterSet [1..*]: MD_CharacterSetCode	dataset & series	[0..*] C / ISO/IEC 10646-1 not used?	-	-	ISO Core element
41	+ topicCategory [1..*] : MD_TopicCategoryCode	dataset & series	C / if hierarchyLevel equals “dataset”?	-	Topic category [1..*]	ISO Core element
45	+ extent [1..*] : EX_Extent	dataset & series	C / if hierarchyLevel equals “dataset”? either extent.geographic Element.EX_ GeographicBoundin g Box or extent.geographic Element.EX_ Geographic Description is required	-	Geographic location [1..*]	INSPIRE SC10: There is at least one instance defining the geographic location of the resource as a geographic bounding box. **Refer to Dictionary Table:** **A 9 - EX_Extent**
46	+ supplementalInformation [0..1] : CharacterString	dataset & series	[0..1]	-	-	

47	SV_ServiceIdentification					
19119 Table C.1 / 1	+ serviceType [1] : GenericName	service	[1]	-	[1] Spatial data service type	INSPIRE Guidance: Use one of these neutral names: • discovery • view • download • transformation • invoke • other
19119 Table C.1 / 2	+ serviceTypeVersion [0..*] : CharacterString	service	[0..*]	-		
19119 AMD1 Table C.1 / 1	+ extent [0..*] : EX_Extent	service	[0..*] C/ if couplingType equals “mixed” or “tight” either extent.geographicElement.EX_GeographicBoundingBox or extent.geographicElement.EX_GeographicDescription is required	-	[1]	**Refer to Dictionary Table: A 9 - EX_Extent**
19119 AMD1 Table C.1 / 2	+ coupledResource [0..*] : SV_CoupledResource	service	[0..*]	-	-	
19119 AMD1 Table C.5 / 1	+ operationName [1] : CharacterString	service	[1]	-	-	
19119 AMD1 Table C.5 / 2	+ identifier [1] : CharacterString	service	[1]	-	-	
CSW 2.0.2 ISO AP	+ gco:ScopedName [0..1] : CodeType	service	[0..1]	-	-	

19119 AMD1 Table C.1 / 3	+ couplingType [1] : SV_CouplingType	service	[1]	-	-	INSPIRE Guidance: The value is… a. loose if there is no coupled Resource (the operatesOn property of SV_ServiceIdentification is not instantiated); b. tight if the service only operates on the Coupled Resources; c. mixed if the service operates on the Coupled Resources and external dataset and dataset series.
19119 Table C.1 / 8	+ containsOperations [1..*] : SV_OperationMetadata	service	[1..*]	-	-	
19119 Table C.2 / 1	+ operationName [1] : CharacterString	service	[1]	-	-	INSPIRE Guidance: Default value is unknown (a unique identifier for this interface)
19119 Table C.2 / 2	+ DCP [1..*] : DCPlist	service	[1..*]	-	-	INSPIRE Guidance: Default value is WebServices.
19119 Table C.2 / 3	+ operationDescription [0..1] : CharacterString	service	[0..1]	-	-	
19119 Table C.2 / 6	+ connectPoint [1..*] : CL_OnlineResource	service	[1..*]	-	-	(handle for accessing the service interface)
>>397	+ linkage [1] : URL	service	[1]	-	[1]	
19119 Table C.1 / 9	+ operatesOn [0..*] : MD_DataIdentification (C)	service	[0..*]	-	Coupled resource (C)	INSPIRE SC11: This instance shall be instantiated by reference (e.g. xlink). INSPIRE Guidance: Conditional to services, Mandatory if linkage to datasets on which the service operates are available.

A 6. CI_Citation (for MD_Identification.citation)

#	NETMAR Metadata Profile	NETMAR MD scope	ISO 19115/19119 analysis	InterRisk analysis	INSPIRE analysis	Comments
359	CI_Citation					This table is for instance MD_Metadata. identificationInfo[1].MD_Identification.citation
360	+ title [1] : CharacterString	dataset & series, service	[1]	-	Resource title [1]	
361	+ alternateTitle [0..*] : CharacterString	dataset & series, service	[0..*]	-	-	
362	+ date [1..*] : CI_Date	dataset & series, service	[1..*]	-	Temporal reference: Date of publication (C), Date of last revision (C), Date of creation (C)	ISO Core element. INSPIRE Guidelines: Conditional, at least one temporal reference is required (i.e. #24/362 or #337). INSPIRE SC7: There shall not be more than one instance declared as a creation date. INSPIRE Guidelines: To be compliant with the INSPIRE Data Specifications - Guidelines, it is recommended, if feasible, to report the date of last revision.
>394	+ date [1] : Date	dataset & series, service	[1]	-	-	
>395	+ dateType [1] : CI_DateTypeCode	dataset & series, service	[1]	-	-	
365	+ identifier [0..*] : MD_Identifier (| RS_Identifier)	dataset & series, service	[0..*]	-	Unique resource identifier (M)	INSPIRE SC8: Mandatory for spatial dataset and spatial dataset series.
>207	+ code [1] : CharacterString	dataset & series, service	[1]	-	-	
>208.1	+ codeSpace [0..1] : CharacterString	dataset & series, service	[1]	-	-	Requires RS_Identifier subclass.
367	+ citedResponsibleParty [0..*] : CI_ResponsibleParty	dataset & series, service	[0..*]	-	-	Definition: name and position information for an individual or organization that is responsible for the resource
369	+ series [0..1] : CI_Series	dataset & series, service	[0..1]	-	-	

> 404	+ name [1] : CharacterString	dataset & series, service	[0..1]	-	-	

A 7. MD_Keywords

#	NETMAR Metadata Profile	NETMAR MD scope	ISO 19115/19119 analysis	InterRisk analysis	INSPIRE analysis	Comments
52	MD_Keywords		Lines 53-55			
53	+ keyword [1..*] : CharacterString	dataset & series, service	[1..*]	-	Keyword value (M)	
54	+ type [0..1] : MD_KeywordTypeCode	dataset & series, service	[0.1]	-	-	
55	+ thesaurusName [0..1] : CI_Citation (C)	dataset & series, service	[0..1]	-	Originating controlled vocabulary (C)	INSPIRE Guidelines: Conditional, mandatory if the keyword value originates from a controlled vocab. NETMAR Guidelines: Use same condition rules as INSPIRE.
>360	+ title [1] : CharacterString	dataset & series, service	[1]	-	-	
>361	+ alternateTitle [0..*] : CharacterString	dataset & series, service	[0..*]	-	-	
>362	+ date [1..*] : CI_Date	dataset & series, service	[1..*]	-	-	
>>394	+ date [1] : Date	dataset & series, service	[1]	-	-	
>>395	+ dateType [1] : CI_DateTypeCode	dataset & series, service	[1]	-	-	

A 8. MD_Resolution

#	NETMAR Metadata Profile	NETMAR MD scope	ISO 19115/19119 Analysis	InterRisk analysis	INSPIRE analysis	Comments
59	MD_Resolution		Lines 60-61			
60	+ equivalentScale [0..1] : MD_RepresentativeFraction	dataset & series, service	[0..1] C / distance not documented?	-	-	
>57	+ denominator [1] : Integer	dataset & series, service	[1]	-	-	
61	+ distance [0..1] : Distance	dataset & series, service	[0..1] C / equivalentScale not documented?	-	-	

A 9. EX_Extent

#	NETMAR Metadata Profile	NETMAR MD scope	ISO 19115/19119 Analysis	InterRisk analysis	INSPIRE analysis	Comments
334	EX_Extent		Lines 335-338			
335	+ description [0..*] : CharacterString	dataset & series, service	[0..1] C / geographicElement and temporalElement and verticalElement not documented?	-	-	
336	+ geographicElement [0..*] : EX_GeographicDescription	dataset & series, service	[0..*] C / description and temporalElement and verticalElement not documented?	-	-	Using subclass EX_GeographicDescription
>349	+ geographicIdentifier [1] : MD_Identifier (|RS_Identifier)	dataset & series, service	[1]	-	-	
336	+ geographicElement [0..*] : EX_GeographicBoundingBox (C)	dataset & series, service	[0..*] C / description and temporalElement and verticalElement not documented?	-	Geographic bounding box [0..*] Conditional, see comment	Using subclass EX_GeographicBoundingBox INSPIRE Guidelines: Conditional, mandatory for spatial dataset and dataset series. Conditional for spatial services, Mandatory for services with an explicit geographic extent. NETMAR Guidelines: Use same condition rules as INSPIRE.
>344	+ westBoundLongitude [1] : Decimal	dataset & series, service	[1]	-	-	
>345	+ eastBoundLongitude [1] : Decimal	dataset & series, service	[1]	-	-	
>346	+ southBoundLatitude [1] : Decimal	dataset & series, service	[1]	-	-	
>347	+ northBoundLatitude [1] : Decimal	dataset & series, service	[1]	-	-	

337	+ temporalElement [1..*] : EX_TemporalExtent	dataset & series, service	C / description and geographicElement and verticalElement not documented?	-	Temporal reference: Temporal extent (C)	INSPIRE Guidelines: Conditional, at least one temporal reference is required (i.e. #24/362 or #337). ISO Core element (Conditional: EX_TemporalExtent or EX_VerticalExtent)
>351	+ extent [1] : TM_Primitive	dataset & series, service	[1]	-	-	
338	+ verticalElement [0..*] : EX_VerticalExtent	dataset & series, service	C / description and geographicElement and temporalElement not documented?	-	-	ISO Core element (Conditional: EX_TemporalExtent or EX_VerticalExtent)
>355	+ minimumValue [1] : Real	dataset & series, service	[1]	-	-	
>356	+ maximumValue [1] : Real	dataset & series, service	[1]	-	-	
>358	+ verticalCRS [1] : SC_CRS	dataset & series, service	[1]	-	-	

A 10. MD_Distribution

#	NETMAR Metadata Profile	NETMAR MD scope	ISO 19115/19119 Analysis	InterRisk analysis	INSPIRE analysis	Comments
270	MD_Distribution		Lines 271-273			
271	+ distributionFormat [1..N] : MD_Format	dataset & series, service	[0..*] C / MD_Distributor. distibutorFormat not documented?	-	Encoding Conditional, see comment	INSPIRE note: Mandatory for most INSPIRE data specifications concerning dataset and dataset-series. **Refer to Dictionary Table:** **A 11 - MD_Format**
272	+ distributor [0..*] : MD_Distributor	dataset & series, service	[0..*]	-	-	
>280	+ distributorContact [1] : CI_ResponsibleParty	dataset & series, service	[1]	-	-	
>282	+ distributorFormat [0] : MD_Format	dataset & series, service	[0..N] C / MD_Distribution. distributionFormat not documented?	-	-	Encode MD_Format info in MD_Distribution. distributionFormat (271) as required by INSPIRE data specifications. **Refer to Dictionary Table:** **A 11 - MD_Format**

273	+ transferOptions [0..*] : MD_DigitalTransferOptions (C)	dataset & series, service	[0..*]	-	Resource locator Conditional, see comment	INSPIRE Guidelines: For dataset and dataset series, mandatory if a URL is available to obtain more information on the resources and/or access related services. For services, mandatory if linkage to the service is available. INSPIRE Guidelines: • Specify a valid URL to the resource. If no direct link to a resource is available, provide link to a contact point where more information about resource is available. • For a service, the Resource Locator might be one of the following: - A link to the service capabilities document; - A link to the service WSDL document (SOAP Binding); - A link to a web page with further instructions - A link to a client application that directly accesses the service NETMAR Guidelines: Use same condition rules as INSPIRE.
>276	+ transferSize [0..1] : Real	dataset & series, service	[0..1]	-	-	Note: size is expressed in megabytes.
>277	+ online [1] : CI_Onlineresource	dataset & series, service	[1]	-	[1]	ISO Core element.
>>397	+ linkage [1] : URL	dataset & series, service	[1]	-	[1]	
>>398	+ protocol [0..1] : CharacterString	dataset & series, service	[1]	-	-	
>>401	+ description [0..1] : CharacterString	dataset & series, service	[1]	-	-	

>>402	+ function [0..1] : CI_OnLineFunctionCode	dataset & series, service	[1]	-	-	Note: This codelist is being extended in the proposed revised ISO 19115. Consider this proposed updated code list for NETMAR use: + download + information + offlineAccess + order + search «new» + completeMetadata + browseGraphic + upload + emailService + browsing + fileAccess

A 11. MD_Format

#	NETMAR Metadata Profile	NETMAR MD scope	ISO 19115/19119 Analysis	InterRisk analysis	INSPIRE analysis	Comments
284	MD_Format		Lines 285-290			
285	+ name [1] : CharacterString	dataset & series, service	[1]	-	[1]	INSPIRE example: e.g. “Protected sites GML application schema”.
286	+ version [1] : CharacterString	dataset & series, service	[1]	-	[1]	INSPIRE example: e.g. “version 3.1; GML, version 3.2.1”.
288	+ specification [1] : CharacterString	dataset & series, service	[0..1]	-	[1]	INSPIRE example: e.g. “D2.8.I.9 Data Specification on Protected sites – Guidelines”.
289	+ fileDecompressionTechnique [0..1] : CharacterString	dataset & series, service	[0..1]	-	-	

A 12. DQ_DataQuality

#	NETMAR Metadata Profile	NETMAR MD scope	ISO 19115/19119 Analysis	InterRisk analysis	INSPIRE analysis	Comments
78	DQ_DataQuality		Lines 79-81			
79	+ scope [1] : DQ_Scope	dataset & series, service	[1]	-	-	
>139	+ level [1] : MD_ScopeCode	dataset & series, service	[1]	-	-	
>141	+ levelDescription [1] : MD_ScopeDescription	service	[1]	-	-	
80	+ report [0..*] : DQ_Element	dataset & series, service	[0..*] C / lineage not provided?	-	Conformity [0..*]	INSPIRE Guidance: INSPIRE defines three degrees of conformity: conformant, not conformant and not evaluated. The expression of these three degrees follows the following rules: • When the conformity to an INSPIRE Specification has been evaluated, it shall be reported as a domain consistency element (i.e. an instance of DQ_DomainConsistency) in ISO 19115 metadata. In that case, if the evaluation has passed, the degree is conformant, otherwise it is not conformant. • The absence of ISO 19115 metadata related to the conformity to an INSPIRE specification does imply that the conformity to this specification has not been evaluated. INSPIRE Guidance: There may be more than one INSPIRE conformity report.
>100	+ nameOfMeasure [0..1] : CharacterString	dataset & series, service	[0..*]	-	(C, see note)	INSPIRE Guidance: Example data specification example guidance; Refer to INSPIRE data specifications for appropriate reporting name, if applicable.

>101	+ measureIdentification [0..1] : MD_Identifier (| RS_Identifier)	dataset & series, service	[0..1]	-	-	INSPIRE Guidance: This metadata element of ISO 19115 will contain the identifier of the conformity statement. This identifier will be used by the application to differentiate the conformance statement related to INSPIRE from others. (However, no formal practical INSPIRE guidance on this to date)
>>207	+ code [1] : CharacterString	dataset & series, service	[1]	-	-	e.g. “Conformity_001”
>>208.1	+ codeSpace [0..1] : CharacterString	dataset & series, service	[0.1]	-	-	Requires RS_Identifier subclass. e.g. “INSPIRE”
>103	+ evaluationMethodType [0..1] : DQ_EvaluationMethodTypeCode	dataset & series, service	[0..1]	-	-	
>104	+ evaluationMethodDescription [0..1] : CharacterString	dataset & series, service	[0..1]	-	-	INSPIRE Guidance: Example data specification example guidance; If the reported data quality results are derived or aggregated (i.e. the scope levels for evaluation and reporting are different), the derivation or aggregation should also be specified using this property.
>105	+ dateTime [0..*] : CI_Citation	dataset & series, service	[0..*]	-	-	INSPIRE Guidance: Example data specification example guidance; This should be date or range of dates on which the data quality measure was applied.
>107	+ result[1] : DQ_Result	dataset & series, service	[0..1]	-	-	INSPIRE Guidance: Use DQ_ConformanceResult subclass for INSPIRE conformity result
>>130	+ specification [1] : CI_Citation	dataset & series, service	[1]	-	Specification [1]	
>>>360	+ title [1] : CharacterString	dataset & series, service	[1]	-	-	e.g. title: “INSPIRE Data Specification on orthoimagery - Guidelines”
>>>362	+ date [1..*] : CI_Date	dataset & series, service	[1..*]	-	-	
>>>>394	+ date [1] : Date	dataset & series, service	[1]	-	-	
>>>>395	+ dateType [1] : CI_DateTypeCode	dataset & series, service	[1]	-	-	

>>131	+ explanation [1] : CharacterString	dataset & series, service	[1]	-	-	INSPIRE Guidance: ISO 19115 mandates an explanation of the meaning of the conformance for this result. A default explanation such as “See the referenced specification” can be used.
>>132	+ pass [1] : Boolean	dataset & series, service	[1]	-	Degree [1]	INSPIRE Guidance: The first two degrees of conformity defined in the INSPIRE Implementing rules for metadata map to two values of the Boolean domain of ISO 19115. The last value corresponds to the case where no conformance statement is expressed in the metadata for the related specification.
81	+ lineage [1] : LI_Lineage + lineage [0..1] : LI_Lineage	dataset & series service	[0..1] C / report not provided?	-	Lineage (C)	INSPIRE Guidance: • Mandatory for spatial dataset and spatial dataset series. • Not applicable to services.
>83	+ statement [1] : CharacterString	dataset & series service	C / (DQ_DataQuality. scope. DQ_Scope.level = “dataset” or “series”)?	-	-	

A 13. MD_Constraints (for MD_Identification.resourceConstraints)

#	NETMAR Metadata Profile	NETMAR MD scope	ISO 19115/19119 Analysis	InterRisk analysis	INSPIRE analysis	Comments
67	MD_Constraints		Line 68			This table is for instance MD_Metadata. identificationInfo[1].MD_Identification. resourceConstraints Can use MD_Constraints or one of its two subclasses: MD_LegalConstraints or MD_SecurityConstraints.
68	+ useLimitation [0..*] : CharacterString	dataset & series, service	[0..*]	-	Conditions applying to access and use [0..*] Conditional: Refer to # 35	INSPIRE legislation: If no conditions apply to the access and use of the resource, 'no conditions apply' shall be used. If conditions are unknown, 'conditions unknown' shall be used. INSPIRE legislation: This element shall also provide information on any fees necessary to access and use the resource, if applicable, or refer to a uniform resource locator (URL) where information on fees is available.
69	MD_LegalConstraints		Lines 70-72 and 68			Subclass of MD_Constraints
70	+ accessConstraints [0..*] : MD_RestrictionCode	dataset & series	[0..*]	-	Limitations on public access [0..*] Conditional: Refer to # 35	
72	+ otherConstraints [0..*] : CharacterString	dataset & series	[0..*] C / accessConstraints or useConstraints equal "otherRestrictions"?	-	Limitations on public access [0..*] Conditional: Refer to # 35	INSPIRE legislation: If there are no limitations on public access, this metadata element shall indicate that fact.
73	MD_SecurityConstraints		Lines 74-77 and 68			Subclass of MD_Constraints

74	+ classification [1] : MD_ClassificationCode	dataset & series	[1]	-	Limitations on public access [0..*] Conditional: Refer to # 35	

A 14. MD_Constraints (for metadataConstraints)

#	NETMAR Metadata Profile	NETMAR MD scope	ISO 19115/19119 Analysis	InterRisk analysis	INSPIRE analysis	Comments
67	MD_Constraints		Line 68			This table is for instance MD_Metadata. metadataConstraints. Can use MD_Constraints or one of its two subclasses: MD_LegalConstraints or MD_SecurityConstraints. NETMAR Guidance: Refer to Dictionary Table A 13 MD_Constraints (for MD_Identification.resourceConstraints) for guidance (i.e. metadataConstraints is not part of INSPIRE legislation, but we can follow the same implementation guidance if required).
68	+ useLimitation [0..*] : CharacterString	dataset & series, service	[0..*]	-	-	
69	MD_LegalConstraints		Lines 70-72 and 68			Subclass of MD_Constraints
70	+ accessConstraints [0..*] : MD_RestrictionCode	dataset & series	[0..*]	-	-	
72	+ otherConstraints [0..*] : CharacterString	dataset & series	[0..*] C / accessConstraints or useConstraints equal “otherRestrictions”?	-	-	
73	MD_SecurityConstraints		Lines 74-77 and 68			Subclass of MD_Constraints
74	+ classification [1] : MD_ClassificationCode	dataset & series	[1]	-	-	