



Preparatory Phase for the pan-European  
Research Infrastructure DANUBIUS-RI  
“The International Centre for advanced  
studies on river-sea systems”

# Implementation of application interfaces

## Deliverable 8.15



European  
Commission

This project has received funding from the European Union's  
Horizon 2020 Research and Innovation Programme under  
Grant Agreement No 739562

<b>Project Full title</b>	Preparatory Phase for the pan-European Research Infrastructure DANUBIUS-RI “The International Centre for advanced studies on river-sea systems”
<b>Project Acronym</b>	DANUBIUS-PP
<b>Grant Agreement No.</b>	739562
<b>Coordinator</b>	Dr. Adrian Stanica
<b>Project start date and duration</b>	
<b>Project website</b>	www.danubius-pp.eu

<b>Deliverable Nr.</b>	8.15	<b>Deliverable Date</b>	M36
<b>Work Package No.</b>	8		
<b>Work Package Title</b>	ICT e-Infrastructure and digital data cloud storage		
<b>Responsible</b>	ROEDU & UPC		
<b>Authors &amp; Institutes Acronyms</b>	<p>SPAIN</p> <p><b>Port Authority of Seville (POS)</b></p> <p><b>Fundación de Investigación de la Universidad de Sevilla (FIUS), and University of Sevilla (USE), on behalf of POS</b></p> <p>Albert Liñán Maho (USE)</p> <p>Sheila Izquieta Rojano (FIUS)</p> <p>Manuel A. Moreno García (FIUS)</p> <p>Antonio Torralba Silgado (FIUS)</p> <p>Antonio Bejarano Moreno (POS)</p>		
<b>Status:</b>	<b>Final (F)</b>	●	
	<b>Draft (D)</b>		
	<b>Revised draft (RV)</b>		
<b>Dissemination level:</b>	<b>Public (PU)</b>	●	
	<b>Restricted to other program participants (PP)</b>		

	<b>Restricted to a group specified by the consortium (RE)</b>	
	<b>Confidential, only for members of the consortium (CO)</b>	

## Table of contents

1. Executive summary / abstract .....	7
2. Introduction .....	8
3. Danubius services.....	9
4. Web services standard technologies.....	11
4.1. Types of web services .....	12
4.1.1. SOAP (Simple Object Access Protocol).....	12
4.1.2. REST (Representational State Transfer).....	14
4.1.2.1. REST Advantages .....	15
4.1.3. REST and SOAP comparison .....	16
5. Secure access to applications and services .....	17
5.1. SAML.....	18
5.2. OAuth .....	20
5.3. Open ID connect.....	20
5.4. Edugain .....	21
5.5. KEYCLOAK .....	23
6. Standards for data access .....	24
6.1. JSON.....	26
6.2. XML .....	27
6.3. WDSL .....	28
6.4. NetCDF.....	29
7. Standards for Metadata access.....	30
7.1. Dublin Core.....	31
7.2. Darwin core (biodiversity).....	32
7.3. EML, Ecological Metadata Language (environmental data) .....	32
8. Supersites survey .....	33
9. Recommendations from D8.2 .....	34
9.1. The Open Geospatial Consortium (OGC) .....	34
10. Selection of datasets for e-services DEMONstration.....	36
11. Selection of tools for e-services DEMONstration.....	37
11.1. R .....	37
11.2. Rstudio.....	38
11.3. Tidyverse .....	39

11.3.1.	ggplot2 .....	39
11.3.2.	dyplr .....	39
11.3.3.	tidyr .....	39
11.4.	Shiny .....	40
11.5.	Flexdashboard .....	40
11.6.	Leaflet .....	41
11.7.	NetCDF4 .....	42
11.8.	DT(DataTable) .....	43
11.9.	Rmarkdown .....	43
12.	User interface DEMO .....	45
12.1.	Requirements .....	45
12.1.1.	Responsive .....	46
12.1.2.	Visual/Interactive Design and Information Structure .....	47
12.2.	Overview .....	48
12.2.1.	Layout .....	48
12.2.2.	Pages .....	50
12.2.2.1.	Description .....	50
12.2.2.2.	River Overview .....	50
12.2.2.3.	River Sections .....	50
12.2.2.4.	River Nodes .....	51
12.2.2.5.	Tidal Tables .....	51
12.2.2.6.	Data & Metadata .....	51
12.2.2.7.	Training .....	52
12.2.2.8.	Publications .....	52
13.	Conclusions .....	53
14.	References .....	54

## List of figures

Figure 1: SSO in SAML .....	20
Figure 2: OpenID connect Architecture .....	21
Figure 3: Who is eduGAIN for? .....	22
Figure 4: eduGAIN schema .....	23
Figure 5: Keycloak main features .....	24
Figure 6: JSON example .....	27

Figure 7: XML example .....	28
Figure 8: NetCDF eststructure example.....	30
Figure 9: DANUBIUS partners Interactive Map .....	34
Figure 10. Web services framework of OGC geoprocessing standards (OSGeoLive, 2018 .....	36
Figure 11: R logo .....	37
Figure 12: Shiny screenshot .....	41
Figure 13: Leaflet Screenshot .....	42
Figure 14: DT Screenshot.....	43
Figure 15: Rmarkdown example Code vs Visualization.....	44
Figure 16: Responsive Design .....	47
Figure 17: Layout Image 1 .....	49
Figure 18: Layout Image 2 .....	49
Figure 19: Layout Image 3 .....	50
Figure 20: DEMO Data & Metadata.....	51
Figure 21: DEMO list of Publications.....	52

## List of tables

Table 1: DANUBIUS-RI services according to WP2 .....	10
Table 2: SOAP, advantages and disadvantages .....	14
Table 3: HTTP Methods .....	14
Table 4: REST and SOAP main features .....	17
Table 5. Recommended data formats in DANUBIUS-RI (DANUBIUS-PP D7.3).....	25
Table 6. Data classification by sampling form, area and type of measurement for Nodes. ....	26
Table 7. List of standards of OGC in alphabetical order. Services are in yellow. ....	35

## 1. Executive summary / abstract

### What is the focus of this Deliverable?

This report focuses on the implementation of interfaces and their most important requirements. Additionally, and as a supplementary document, a service integration template is provided. This template will be used to collect information from each partner in order to implement its services on the platform. Moreover, as a third pillar of this deliverable, we present a DEMO that shows how the interface could work.

This deliverable is mainly based on the information gathered in deliverables 8.9 and 8.10, also led by POS. It follows the line of deliverable 8.14 (draft) and completes the information presented on it. The abovementioned DEMO has been developed with one of the Guadalquivir Estuary Supersite datasets, which is made up of two models of finite elements of the Guadalquivir river. The development tools, operation and specifications of the DEMO are explained in this deliverable as well.

In summary, this deliverable consists of 3 files:

- The report. Document that gathers all the ideas associated with the development of an interface: standards, context, requirements, security, programming languages, etc.
- The Service Integration Template: its function is to facilitate the proper integration of services provided by the different DANUBIUS partners. It places great emphasis on standard formats and metadata management through FAIR principles.
- The DEMO interface: interactive platform that allows users to access DANUBIUS services. Aimed at data models, virtual labs, temporary data and repositories. Composed of graphics, dynamic tables, text, images, ...

### What are the Deliverable contents?

This document is divided into the following sections:

1. Executive summary / abstract
2. Introduction
3. Danubius services
4. Web services standard technologies
5. Secure access to applications and services

6. Standards for data access
7. Standards for Metadata access
8. Supersites survey
9. Recommendations from D8.2
10. Selection of datasets for e-services DEMOnstration
11. Selection of tools for e-services DEMOnstration
12. User interface DEMO
13. Conclusions
14. References

## 2. Introduction

DANUBIUS-RI will be a pan-European distributed research infrastructure dedicated to interdisciplinary studies of large river-sea (RS) systems. It will enable and support research addressing the conflicts between society's demands, environmental change and environmental protection in river-sea systems worldwide. DANUBIUS-RI will be a distributed RI that will collaborate with, and enhance, existing European research infrastructure and programmes.

The main objectives of WP8 are to offer the necessary computing capacity and to provide the necessary tools for the development of all connectivity and calculation elements for DANUBIUS-RI.

The first objective determines the storage and communication infrastructure and simulation tools for the distributed Research Infrastructure (RI) according to the requirements for each component, focusing on the connectivity between the Data Centre, the Hub, the Nodes and Supersites and users. The secondary objective affords the necessary tools and techniques for the development of all computing and connective elements (research, development, innovation and education oriented), including High Performance and Cloud computing, and application interfaces, such as an adaptive Virtual Research Environment, which will be in synergy with other on-going pan-European Research Infrastructures developments and on-going EC Directives in this regard.

This report presents the implementations of the interfaces and services and provides templates for associated organisations to provide compatible and standardised services that intermesh with those of the core. The recommendations from Task 8.2 provide the foundation, while taking account of the capabilities, existing facilities and preferences of contributing partners. A DEMOnstration of these interfaces will serve to enable the ICT e-infrastructure and e-learning centre described in T5.8.,



including a virtual research environment for research applications, development and innovation as well as education applications.

As mentioned above, in this report, in addition to showing how an interface works, it is presented a general template for service integration. The template explains the different fields that are considered necessary to develop an interface and the standards that must be followed. It is focused on data and metadata formats, and provides some recommendations for the application of FAIR principles. The template also gives some information about terms of use and licenses.

The importance of standards is one of the main issues of this deliverable. Below, some concepts are briefly explained to facilitate the understanding of the rest of the document.

Data standards are a very important part of improving data quality for better decision making in environmental management programs. The use of data standards enables reusability of data elements and their metadata that can reduce redundancy between systems, thereby improving reliability and often reducing cost.

A metadata standard is a requirement which is intended to establish a common understanding of the meaning or semantics of the data, to ensure correct and proper use and interpretation of the data by its owners and users.

As presented in deliverable 8.10; the FAIR Data Principles are a set of guiding principles in order to make data findable, accessible, interoperable and reusable. These principles provide guidance for scientific data management and stewardship and are relevant to all stakeholders in the current digital ecosystem. They directly address data producers and data publishers to promote maximum use of research data. Research libraries can use the FAIR Data Principles as a framework for fostering and extending research data services.

### **3. Danubius services**

DANUBIUS-RI is supposed to support research community by supplying services and tools to improve knowledge and e-learning opportunities.

In this sense, the following services are expected to be provided in DANUBIUS-RI according to WP2 recommendations:

ID	Service Name	Service Description
1.	Computing as a Service	Run a model simulation in the cloud / on a supercomputer
2.	Software as a Service	Simulation software (versioned) is available in the cloud
3.	Model as a Service	Model applications of river-sea systems are available for modellers (schematisations, time series, calibrated and validated parameter sets)
4.	Data as a Service	Data sets, quality controlled and prepared for modelling purposes are available for modellers (bathymetry, time series, parameter sets, ...)
5.	Research grants	Researchers can work together with the best modelling experts in Europe. After application, the winning grants are selected.
6.	Standardisation and guidelines	DANUBIUS-RI experts define standards and guidelines for data formats, Good Modelling Practice, software development
7.	Software engineering	Integrate new research codes in existing software
8.	Community management	Web portal with a moderated forum for good modelling practice
9.	Workshops and hackathons	Venues where researchers, modellers, developers meet and collaborate on specific topics
10.	Training	Training courses on the use of modelling software, model applications, good modelling practice, model development, etc.
11.	Webinars, MOOC's	Open access digital courses and presentations
12.	Model application	Application of DANUBIUS model applications for a specific purpose by a DANUBIUS-RI expert
13.	Modelling support	Support with the software, datasets and model applications by a DANUBIUS expert
14.	Remote control systems	Get access to (physical) research facilities from a distance

Table 1: DANUBIUS-RI services according to WP2

From the previous services lists we can identify services that could be provided *freely* or as a *public* service where no especial security issues are needed, out of maybe, previous invitation requirements. But there are some other services that only will be provided to specific users.

Related to this second group of users, it is crucial to guarantee secure access to data, software, computing services, etc. This secure access will be provided according to authorization technologies and protocols following the most important standards used nowadays.

## 4. Web services standard technologies

When we are talking about services and e-services, it is important to point out that all these services are provide in a remote way. This means, that all the (e-)services provided by DANUBIUS-RI must be able to accept requests from users located in any place all over the world.

At this point, there are to concepts that deserve to clarify: *user* and *all over the world*.

1. User, in this case, it can be also called as end-user or client; and are referred both to the ultimate human users of a software product or also to a client application that can use in an autonomous ways certain services; for example, an application that automatically read meteorological data from a database and generate warnings depending on the gathered values.
2. When talking to *all over the world*, we are referring to that independently of the hardware, software, way to access to the network, etc. they shall be able to access to the DANUBIUS-RI services in the same way thought standard protocols and processes. Also, it refers that due to the remote nature of the access it is crucial to protect the way to access to the services and data by using security and authentication mechanisms.

The point related to how a user could access to data and services is described in the current chapter, the security aspects and technologies will be described in the next one.

The processes that allow to user to get access to services and data through the Internet and independently from the location are called *Web Services*.

The process of preparing an application to support more than one language and data format is called internationalization. Localization is the process of adapting an internationalized application to support a specific region or locale. Examples of locale-dependent information include messages and user interface labels, character sets and encoding, and date and currency formats. Although all client user interfaces should be internationalized and localized, these processes are particularly important for web applications because of the global nature of the web

**Web services** are client and server applications that communicate over the World Wide Web's (WWW) HyperText Transfer Protocol (HTTP). As described by the World Wide Web Consortium (W3C), web services provide a standard means of interoperating between software applications running on a variety of platforms and frameworks. Web services are characterized by their great interoperability and extensibility, as well as their machine-processable descriptions, thanks to the use of XML. Web services can be combined in a loosely coupled way to achieve complex operations.

Programs providing simple services can interact with each other to deliver sophisticated added-value services.

In other words, A web service is any piece of software that makes itself available over the internet and uses a standardized messaging system. It could be considered a collection of open protocols and standards used for exchanging data between applications or systems. Software applications written in various programming languages and running on various platforms can use web services to exchange data over computer networks like the Internet in a manner similar to communication on a single computer. This interoperability is due to the use of open standards.

#### 4.1. Types of web services

On the conceptual level, a service is a software component provided through a network-accessible endpoint. The service consumer and provider use messages to exchange invocation request and response information in the form of self-containing documents that make very few assumptions about the technological capabilities of the receiver.

On a technical level, web services can be implemented in various ways. The two types of web services discussed in this section can be distinguished as “SOAP” web services and “RESTful” web services

##### 4.1.1. SOAP (Simple Object Access Protocol)

Simple Object Access Protocol (SOAP) was created in 1998 by Dave Winer, Don Box, Bob Atkinson, and Mohsen Al-Ghosein with backing by Microsoft. SOAP is a messaging framework that gained widespread support in the Java, .NET and open source communities during the early part of the 2000s. It has served as the foundation of many Web services projects and provides the mechanism by which many other Web services standards communicate.

SOAP is a way for a program running in one kind of operating system to communicate with a program in the same or another kind of an operating system (such as Linux) by using the World Wide Web's Hypertext Transfer Protocol (HTTP) and its Extensible Markup Language (XML) as the mechanisms for information exchange.

Since Web protocols are installed and available for use by all major operating system platforms, HTTP and XML provide an already at-hand solution to the problem of how programs running under different operating systems in a network can communicate with each other. SOAP specifies exactly how to encode an HTTP header and an XML file so that a program in one computer can call a program in another computer and pass it information. It also specifies how the called program can return a response.

SOAP is often used in conjunction with WSDL (Web Services Description Language). WSDL is a service description language. It describes the interface of the service, i.e., the structure of the XML messages that the service can accept/return. WSDL also describes how those messages are encoded and indicates what protocols the service supports (e.g., SOAP over HTTP).

The SOAP message format and the WSDL interface definition language have gained widespread adoption. Many development tools, such as NetBeans IDE, can reduce the complexity of developing web service applications.

A SOAP-based design must include the following elements.

- A formal contract must be established to describe the interface that the web service offers. WSDL can be used to describe the details of the contract, which may include messages, operations, bindings, and the location of the web service. You may also process SOAP messages in a JAX-WS service without publishing a WSDL.
- The architecture must address complex non-functional requirements. Many web service specifications address such requirements and establish a common vocabulary for them. Examples include transactions, security, addressing, trust, coordination, and so on.
- The architecture needs to handle asynchronous processing and invocation. In such cases, the infrastructure provided by standards, such as Web Services Reliable Messaging (WSRM), and APIs, such as JAX-WS, with their client-side asynchronous invocation support, can be leveraged out of the box.

Advantages	Disadvantages
Using SOAP over HTTP allows for easier communication through proxies and firewalls than previous remote execution technology.	Because of the verbose XML format, SOAP can be considerably slower than competing middleware technologies.
SOAP is versatile enough to allow for the use of different transport protocols (HTTP, SMTP, RSS, etc.).	When relying on HTTP as a transport protocol and not using WS-Addressing or an ESB, the roles of the interacting parties are fixed. Only one party (the client) can use the services of the other. Developers must use polling instead of notification in these common cases
SOAP is platform independent.	Most uses of HTTP as a transport protocol are done in ignorance of how the operation would be modelled in HTTP. Because of this, there is no way to know if the method used (POST, GET, etc.) is appropriate to the operation
SOAP is language independent.	
SOAP is simple and extensible.	

Table 2: SOAP, advantages and disadvantages

4.1.2. REST (Representational State Transfer)

Besides SOAP there is an alternative for the realization of Web services. REST is based on principles, which are used in the largest distributed application - the World Wide Web. Without intention there are many search engines, shops or booking systems that are already available as REST based Web services. The REST architecture is an architecture that describes how the Web should work. REST is neither a product nor a standard

The interface of REST is generic. There is no need for protocol conventions for the communication between client and server. The following list describes the meaning of the HTTP methods and how they are used by REST.

Method	Description
<b>GET</b>	GET queries the representation of a resource. The execution of requests should be free from side effects. GET requests can be sent arbitrarily often. You cannot blame the client for effects caused by GET requests. That means a GET can be sent heedlessly.
<b>POST</b>	With POST you can change the state of a resource. For example you can add a good to a shopping cart. POST isn't free from side effects. For example you can change fields in a data base or start a new process on the server with a POST request.
<b>PUT</b>	New resources can be produced with PUT or you can replace the content of existing resources.
<b>DELETE</b>	Resources can be deleted using DELETE.

Table 3: HTTP Methods

The HTTP methods GET, POST, PUT and DELETE are a generic interface for each REST resource. Database applications have shown that a limited set of commands controlling the lifecycle of data is sufficient. The database query language SQL covers the lifecycle by the SELECT, INSERT, UPDATE and DELETE instructions.

All types of documents can be used as representations for resources. For example HTML, GIF and PDF files are used in the Web. XML is suitable for the transmission of structured data. XML documents can use XLink for references. If you want to realize your application by using Rest, you don't have to learn a new format. You can use well-known formats.

REST messages are self explaining. In a message everything must be contained, in order to interpret the message. There is no knowledge about previous or later messages necessary for the interpretation of a message. The status of an application is represented by the contents of one or several hypertext documents.

REST application are characterized by the following properties:

- Communication takes place on call. The Client is active and requests a representation from the passive server and/or modifies a resource.
- A resource can be addressed by an unique URI.
- The client can request the representation of a resource in form of a document.
- Representations can refer to further resources.
- The server does not monitor the status of its clients. Each query to the server must contain all informations that are necessary for interpreting itself.
- Caching is supported. The server can mark its answers as cacheable or not cacheable.

#### 4.1.2.1. *REST Advantages*

- **Scalability**

The enormous growth of the World Wide Web has shown in which extent web technologies scale. Millions of users use resources, which are offered by many thousand of servers. Proxys and caches increase the performance. Components such as servers, proxys and web applications can separately be installed and attended.

There are lots of formats like HTML, SVG, AVI,... Even new formats can be easily added with MIME types. The size of the transferred documents varies from few bytes to many megabytes.

In the web artificially sessions are generated with cookies and URL rewriting. These sessions are stateful. In this point REST deviates from the web. In REST all interactions are stateless - each operation stands for itself. The representations of the resources contain all necessary information. With HTTP there are no boundaries between the applications. By following a link you can arrive at a completely different application. Since all interactions are stateless, the servers have not to interact. This fact improves the applications' scalability.

- **Access to External Systems**

There is no other application with so many legacy systems like the web. You can access an abundance of other systems via different gateways. The details of other external systems are hidden behind interfaces.

- **Independent Installable Components**

In REST the deployment of the components can be accomplished indepentent by each other. The content of individual web sides can be exchanged without adapting further sides. Independent

deployment is a prerequisite in very large systems like the World Wide Web or the email service of the internet.

- **Composition of services**

It is easy to use individual REST services together. Strictly speaking, there are no REST services. There are only resources. URLs are offering a global address space so documents simply refer to a resource which is in another organization.

#### **4.1.3. *REST and SOAP comparison***

The delivery of SOAP messages is comparable with a company internal mail system. The letter baskets of the coworkers cannot be addressed directly. All letters are delivered at the house's post office. The house's post office must open the letters before and decides then, due to content, to whom the message is passed on. In a REST application the letter baskets are directly addressable. The SOAP specification version 1.2 approaches on REST in this point.

SOAP messages are always addressed to an endpoint, which is implemented by a SOAP router, also called dispatcher. SOAP router is realized through a Servlet or a CGI script. In illustration 3 you can see that the request is delivered to the dispatcher first. Each message is a HTTP POST to the same address.

SOAP is a protocol construction set used to write own application protocols. The protocol describes the structure of the request and the reply. The description of the request and the answer represents a rigid framework that cannot be easily changed. If the server would like to supply further information than so far, then it cannot use the existing interface. A new web service is necessary so that already installed clients remain compatible. Using REST, the server can supply additional information and the interface of the web service can remain unchanged. In SOAP the possibility for the gradual evolution of existing Web services is limited.

The address space between REST and SOAP is a crucial difference. REST uses a global address space with URLs, over which each resource can be addressed. In REST the resource is the centre of attention. There are no services in REST.

REST offers with the GET, POST, PUT and DELETE methods a generic interface. In SOAP all methods for each application must be defined by yourself. Thereby, the development of generic tools and services in the web is obstructed.

REST recommends the use of established standards. For example, the following technologies are at the developer's disposal:



- URLs for addressing
- HTTP methods for communication
- XML, XHTML, HTML, PNG, ... as data formats
- MIME types
- etc.

The standards for HTTP and URIs are central. For the representation you can use arbitrary formats. Future technologies also can be used.

There are some standards like WSDL, UDDI, WS-Security designed exclusively for SOAP applications. REST standards are "real" web standards, used for many purposes, not only for REST.

REST main features	SOAP main features
Few operations with many resources	Many operations with few resources
Focused on scalability and large-scale performance for distributed hypermedia systems	Focused on design of distributed applications.
HTTP GET, HTTP POST, HTTP PUT, HTTP DEL	SMTP, HTTP POST, MQ
XML auto-descriptive	Fixed types, XML Schema
Synchronous	Synchronous y Asynchronous
HTTPS	WS SECURITY
Secure and point to point communication	Secure origin to destination communication

Table 4: REST and SOAP main features

## 5. Secure access to applications and services

This chapter describes authentication process and technologies that will allow to user to reach the data and services provided by DANIBIUS-RI in a secure way for both, users and data/services.

Interfaces for services a data must provide mechanisms not only for Authentication (understood as the process of ascertaining that someone is really who she or he claims to be) but also for Authorization (understood as the process that determines if someone is allowed to take an action or not) in order to manage the access to all the services and resources that will be provided and to track their usage.

AAI (Authentication and Authorization Infrastructure) is an essential point to be satisfied since all the potential users need to be uniquely identified and a set of roles must be assigned to them in order to define which services can be accessed and which not, a number of resources that can be used, etc. To solve this point, different roles can be assigned to different users in order to authorize them or not to perform some actions.

Besides, there are available web technologies that support different AAI mechanisms, and the most important service companies manage their users using open standards. OpenID Connect, OAuth, and SAML are the most extended standards used to access to web-based services, and they can be adapted to almost any potential SaaS-based (Saas, Software as a Service) tool.

AAI mechanisms are also key to support the Federation of resources and services. Some on-going initiatives are already addressing this problem at European level. The AARC project (Authentication and Authorization for Research and Collaboration) and its follow-up AARC2, aim at designing an AAI framework to develop an interoperable infrastructure that supports the researchers to access the services provided by Research Infrastructures, both transversal and community specific. Communities and Research Infrastructures from diverse domains are participating providing feedback and requirements to the architectural design.

One of the main challenges that e-services needs to face to build this new framework for research, is to provide a way to scientists to get access not only to the data but also to the services managing the data lifecycle. Repositories or data portals are the access point where researchers can get the resources they need, including the datasets, but also associated software and even the computing power required to process them. Therefore, these data portals need to integrate an Authentication and Authorization Infrastructure (AAI) capable of managing the user access and also being compatible with other services that handle different stages in the life cycle. The following technologies are quite extended and are integrated into AAI solutions for a cloud environment:

### **5.1. SAML**

The Security Assertion Markup Language is an open standard that defines an XML schema for exchanging authentication and authorization data among services. In general, the parts intervening in this exchange are an Identity Provider (supporting the identification from an entity) and a Service Provider (providing the resources).

Security Assertion Markup Language (SAML) is a standard for logging users into applications based on their sessions in another context. This single sign-on (SSO) login standard has significant advantages over logging in using a username/password:

- No need to type in credentials
- No need to remember and renew passwords
- No weak passwords

Most organizations already know the identity of users because they are logged in to their Active Directory domain or intranet. It makes sense to use this information to log users in to other applications, such as web-based applications, and one of the more elegant ways of doing this is by using SAML.

SAML is very powerful and flexible, but the specification can be quite a handful. OneLogin's open-source SAML toolkits can help you integrate SAML in hours, instead of months. It's come up with a simple setup that will work for most applications.

SAML SSO works by transferring the user's identity from one place (the identity provider) to another (the service provider). This is done through an exchange of digitally signed XML documents.

Consider the following scenario: A user is logged into a system that acts as an identity provider. The user wants to log in to a remote application, such as a support or accounting application (the service provider). The following happens:

1. The user accesses the remote application using a link on an intranet, a bookmark, or similar and the application loads.
2. The application identifies the user's origin (by application subdomain, user IP address, or similar) and redirects the user back to the identity provider, asking for authentication. This is the authentication request.
3. The user either has an existing active browser session with the identity provider or establishes one by logging into the identity provider.
4. The identity provider builds the authentication response in the form of an XML-document containing the user's username or email address, signs it using an X.509 certificate, and posts this information to the service provider.
5. The service provider, which already knows the identity provider and has a certificate fingerprint, retrieves the authentication response and validates it using the certificate fingerprint.
6. The identity of the user is established, and the user is provided with app access

The diagram below illustrates the single sign-on flow for service provider-initiated SSO, i.e. when an application triggers SSO.

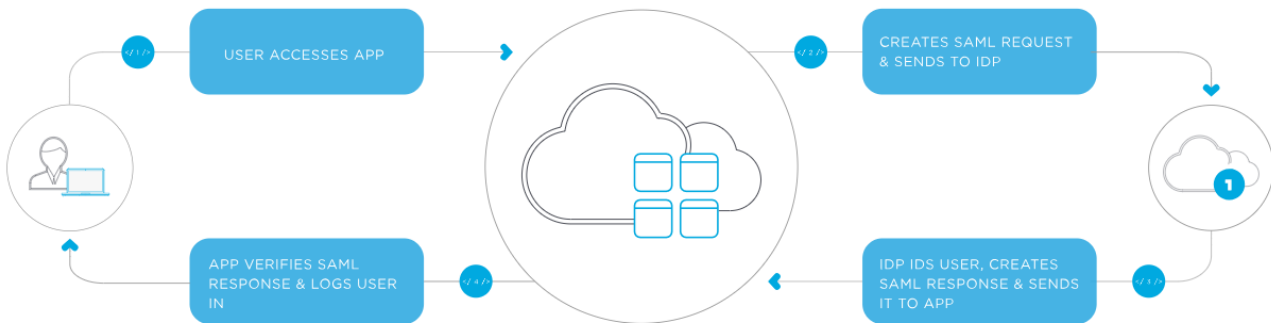


Figure 1: SSO in SAML

## 5.2. OAuth

OAuth is an open standard for authorization, commonly used as a way for Internet users to log into third-party websites using their Microsoft, Google, Facebook or Twitter accounts without exposing their password. Generally, OAuth provides to clients a “secure delegated access” to server resources on behalf of a resource owner. It specifies a process for resource owners to authorize third-party access to their server resources without sharing their credentials. Designed specifically to work with Hypertext Transfer Protocol (HTTP),

OAuth essentially allows access tokens to be issued to third-party clients by an authorization server, with the approval of the resource owner. The third party then uses the access token to access the protected resources hosted by the resource server.

## 5.3. Open ID connect

OpenID Connect is an interoperable identity layer implemented on top of the OAuth 2.0 family of specifications. Since it is built on OAuth 2.0, it standardizes the use of this protocol for login and adds a Discovery process that enables Identity Provider lookup. OpenID Connect allows clients of all types, including Web-based, mobile, and JavaScript clients, to request and receive information about authenticated sessions and end-users. The specification suite is extensible, allowing participants to use optional features such as encryption of identity data, a discovery of OpenID Providers, and session management.

OpenID Connect allows clients of all types, including Web-based, mobile, and JavaScript clients, to request and receive information about authenticated sessions and end-users. The specification suite is extensible, allowing participants to use optional features such as encryption of identity data, discovery of OpenID Providers, and session management, when it makes sense for them.

Most of the popular Internet services or portals (GitHub, Google services, etc.) are compatible with these technologies.

The OpenID Connect architecture is shown in the diagram below:

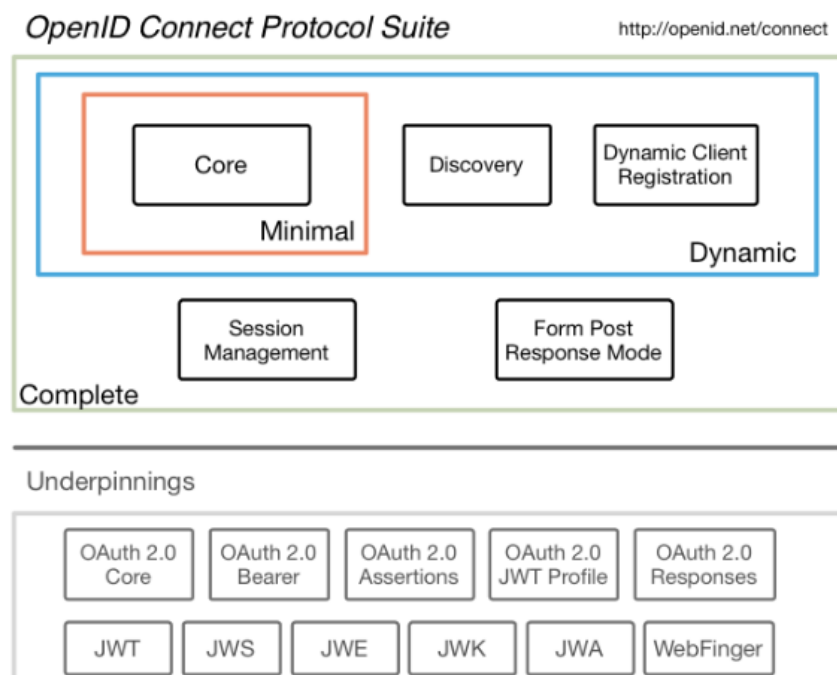


Figure 2: OpenID connect Architecture

#### 5.4. Edugain

This service interconnects research and education identity federations around the world. It is based on trust, so eduGAIN provides the identification from a home institution to another one, to provide access to a member from a third institution.

EduGAIN comprises over 50 participant federations connecting 2,672 Identity Providers to 1,909 Service Providers.

With just one trusted identity provided by user's institution as part of an identity federation participating in eduGAIN, users can access services from other participating federations. Even better, it works with single sign-on (SSO), so that user needs to login only one time during a browser session.

R&E identity federations are typically deployed on a national level and use different architectures, systems and policies. eduGAIN enables identity federations to interconnect, so that participating institutions and services can collaborate without the need to establish individual bilateral connections.

## Who is eduGAIN for?

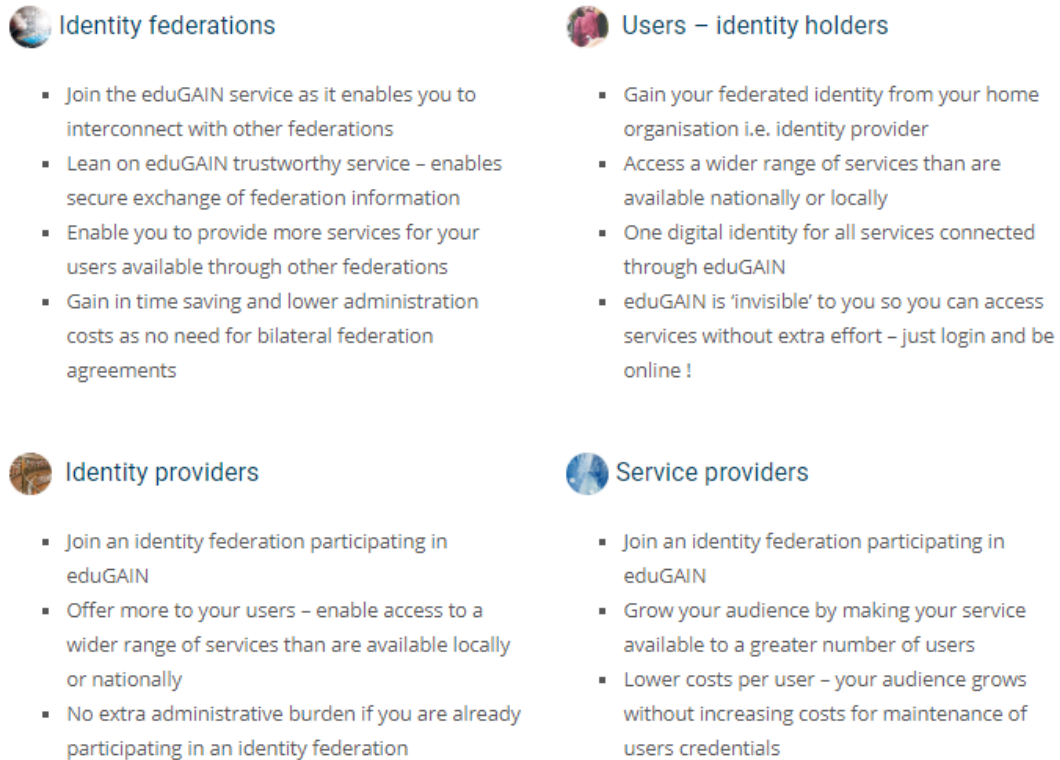


Figure 3: Who is eduGAIN for?

To better understand how eduGAIN works, it is important to keep in mind the following concepts:

- Single identity
- Federated identity environments
- Home organisations
- Identity Provider
- Service Provider
- Identity federation

Many organisations use Authentication and Authorisation Infrastructures (AAs) to build a trusted environment where users can be identified electronically using a **single identity**. The necessity for user identity to cross borders between organisations, domain and services, lead to the creation of **federated identity environments**.

**Home organisations** (e.g. a university, library, research institute etc.) register users by assigning a digital identity. They also operate an **Identity Provider** and are able to authenticate their users and provide a limited set of attributes that characterise the user in a given context. Resource owners

(**Service Providers**) delegate the authentication to Identity Providers in order to control access to the provided resources.

An **Identity federation** is a group of Identity and Service Providers that sign up to an agreed set of policies for exchanging information about users and resources to enable access to and use of the resources. There are many Research and Education identity federations around the globe and they commonly have a national coverage. To discover worldwide identity federations, view the REFEDS map.

eduGAIN interconnects identity federations around the world, simplifying access to content, services and resources for the global research and education community. The eduGAIN technology involves a “metadata service”, which regularly retrieves and aggregates information from participating federations about Service and Identity Providers, and makes this information available to federations.

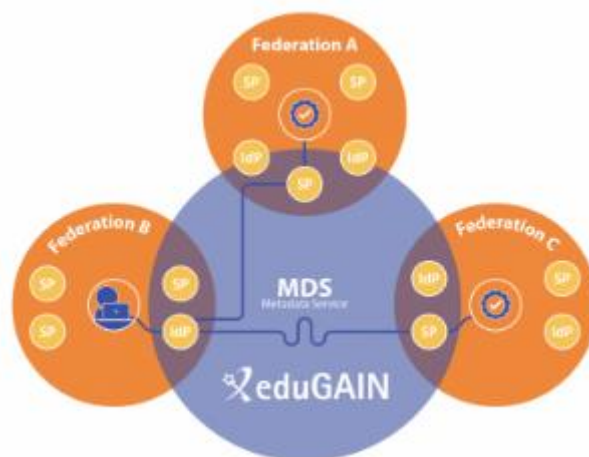


Figure 4: eduGAIN schema

## 5.5. KEYCLOAK

Keycloak is an open source Identity and Access Management solution aimed at modern applications and services. It makes it easy to secure applications and services with little to no code.

Users authenticate with Keycloak rather than individual applications. This means that your applications don't have to deal with login forms, authenticating users, and storing users. Once logged-in to Keycloak, users don't have to login again to access a different application.

This also applied to logout. Keycloak provides single-sign out, which means users only have to logout once to be logged-out of all applications that use Keycloak

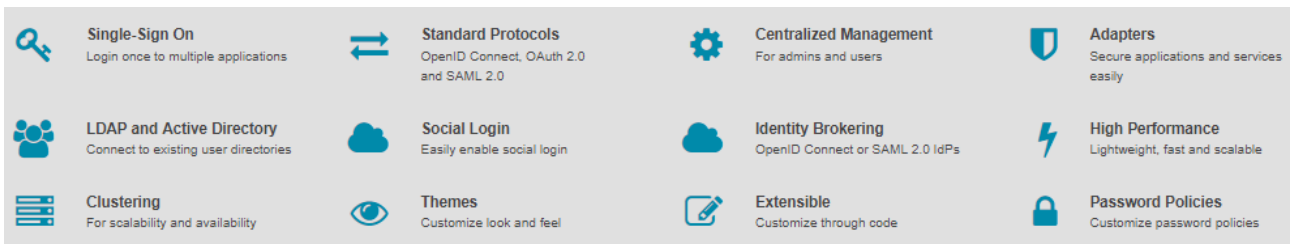


Figure 5: Keycloak main features

Keycloak has built-in support to connect to existing LDAP or Active Directory servers. You can also implement your own provider if you have users in other stores, such as a relational database.

Keycloak is based on standard protocols and provides support for OpenID Connect, OAuth 2.0, and SAML.

## 6. Standards for data access

Data is a key element in the design of any e-service because it determines the type of algorithms to be used within it.

Initially the data inside DANUBIUS-RI will be collected mainly from the Supersites and Nodes. The authors identify 4 main sources of data: (i) from projects carried out at DANUBIUS-RI, (ii) from projects carried out under the DANUBIUS-RI umbrella, (iii) from partners of DANUBIUS-RI and (iv) from repositories and environmental specimen banks that already exist.

Regarding the type of data that will be available/produced within DANUBIUS-RI, the DANUBIUS-PP deliverable 7.3 defines 4 main categories: (i) Observational data, (ii) Experimental data, (iii) Computational outputs and (iv) Compiled data (produced by applying a transformation process on existing data). In parallel, data can be qualitative or quantitative. The data structures identified by the authors are: (i) raster, (ii) vector data, (iii) Gridded or array-oriented data, (iv) hierarchical data and (v) data time series.

The recommended data formats can be seen in Table 5

FORMATS	
Qualitative data	Text files with extension - .XML (text according to an appropriate document type) - .RTF (rich text format), .TXT (plain text data), ASCII - .doc/.docs( widely –used proprietary formats, MS Word), .tex (LaTex )



Quantitative tabular data with extensive metadata	<ul style="list-style-type: none"> <li>--proprietary data formats only if a de facto standard and no international accepted standard exists</li> <li>- Data sets with variable labels</li> <li>- data and metadata inside a relational database</li> <li>-XML file with metadata (WaterML2 etc)</li> </ul>
Quantitative tabular data with minimal metadata	<ul style="list-style-type: none"> <li>- .CSV (comma-separated values), tab (tab-delimited file)</li> <li>- widely-used formats MS Excel (.xls/.xlsx), MS Access (.mdb/.accdb), open documents spreadsheet (.ods)</li> </ul>
Quantitative gridded/array-oriented data, self-describing formats	<ul style="list-style-type: none"> <li>- .nc (netCDF) with CF conventions</li> <li>- .grb (GRIB)</li> <li>- .hdf</li> </ul>
Geospatial data	(vector and raster data), tabular GIS attribute data OGC simple features, proprietary ESRI-shp, ...
Digital audio data	<ul style="list-style-type: none"> <li>- .wav (Waveform Audio Format)</li> <li>- .mp3 (MPEG audio format)</li> </ul>
Digital video data	<ul style="list-style-type: none"> <li>- .MP4 (MPEG 4)</li> <li>- .jp2 (motion JPEG)</li> </ul>
Documentation and scripts	Files in the .RTF, .PDF or .HTM format, plain text or widely used proprietary formats MSWORD(.doc/.docx), MS EXCEL(.xls/.xlsx), SQL, JAVA, JAVASCRIPT

Table 5. Recommended data formats in DANUBIUS-RI (DANUBIUS-PP D7.3).

**Error! Reference source not found.** shows the type of data acquired.

LOCATION	SAMPLING PLATFORM	THEMATIC AREAS	TYPE OF MEASUREMENTS	DATA FORMATS
Analysis Node	Laboratories, research vessels, Autonomous platforms (including UAV)	Biology, Chemistry, Ecotoxicology, Hygiene, Physics of water quality	analytical chemistry, water balance, hydrometry, general water properties, general soil properties, general sediment properties, suspended load properties, dating, biomarker,	Geospatial data (vector and raster data), tabular GIS attribute data  Quantitative tabular data with minimal metadata  Quantitative tabular data with extensive metadata

			hydromorphology evaluation, geodesy, biochemistry, hygiene parameters	Qualitative data Quantitative array-oriented data, self-describing
Observation Node	Earth observation data (satellite , airborne, drone and in-situ)	Biogeochemical data, physical	Ocean colour derived biogeochemical variables (chlorophyll and suspended particulate matter concentrations, dissolved organic matter, other algorithms)  Radiometric and inherent optical properties  Sea surface temperature	Geospatial data (vector and raster data), tabular GIS attribute data  Quantitative tabular data with minimal metadata  Quantitative tabular data with extensive metadata  Qualitative data  Quantitative array-oriented data, self-describing
Modeling Node	Modeled data,	Biology, Imaging, Biogeochemistry, Hydrology, Geology, Chemistry, Ecology	water temperature, salinity, pollution, heavy metals, meteorological marine data	Geospatial data (vector and raster data), tabular GIS attribute data  Quantitative tabular data with minimal metadata  Quantitative tabular data with extensive metadata  Qualitative data  Quantitative array-oriented data, self-describing

Table 6. Data classification by sampling form, area and type of measurement for Nodes.

Common formats for remote sensing data are: GeoTIFF, Hierarchical data format v4 (HDF4), HDF5, netCDF-CF, GRIB. Frequently GIS file formats are: HDF, GeoTIFF, JPEG2000, netCDF, ASCII, MrSid among many others

### 6.1. JSON

JSON (JavaScript Object Notation) is a syntax for storing and exchanging data.

JSON is text, written with JavaScript object notation. Main features:

- stands for JavaScript Object Notation
- is a lightweight data-interchange format
- is "self-describing" and easy to understand
- is language independent, Text can be read and used as a data format by any programming language.

```
{  
  "employees": [  
    {  
      "id": 52626,  
      "name": "Employee One"  
    },  
    {  
      "id": 26565,  
      "name": "Employee Two"  
    }  
  ]  
}
```

Figure 6: JSON example

## 6.2. XML

Extensible Markup Language (XML) is a markup language that defines a set of rules for encoding documents in a format that is both human-readable and machine-readable.

The W3C's XML 1.0 Specification and several other related specifications (all of them free open standards) define XML.

The design goals of XML emphasize simplicity, generality, and usability across the Internet. It is a textual data format with strong support via Unicode for different human languages. Although the design of XML focuses on documents, the language is widely used for the representation of arbitrary data structures such as those used in web services.

Several schema systems exist to aid in the definition of XML-based languages, while programmers have developed many application programming interfaces (APIs) to aid the processing of XML data.

```
<?xml version="1.0" standalone="yes" ?>
<shop location="Birmingham" size="Large">
- <food>
  <Name>Apple</Name>
  <type>fruit</type>
  <cost>15</cost>
</food>
- <food>
  <Name>Carrot</Name>
  <type>vegetable</type>
  <cost>10</cost>
</food>
</shop>
```

Figure 7: XML example

### 6.3. WDSL

Not extensive explanation will be included in this chapter, just a few key point to understand the level of description contained in the WDSL format, due to is commonly used for web services description, especially in SOAP.

Web Services Description Language Version 2.0 (WSDL 2.0) provides a model and an XML format for describing Web services. WSDL 2.0 enables one to separate the description of the abstract functionality offered by a service from concrete details of a service description such as “how” and “where” that functionality is offered.

This specification defines a language for describing the abstract functionality of a service as well as a framework for describing the concrete details of a service description. It also defines the conformance criteria for documents in this language.

WSDL 2.0 describes a Web service in two fundamental stages: one abstract and one concrete. Within each stage, the description uses a number of constructs to promote reusability of the description and to separate independent design concerns.

At an abstract level, WSDL 2.0 describes a Web service in terms of the messages it sends and receives; messages are described independent of a specific wire format using a type system, typically XML Schema.

An operation associates a message exchange pattern with one or more messages. A message exchange pattern identifies the sequence and cardinality of messages sent and/or received as well as who they are logically sent to and/or received from. An interface groups together operations without any commitment to transport or wire format.

At a concrete level, a binding specifies transport and wire format details for one or more interfaces. An endpoint associates a network address with a binding. And finally, a service groups together endpoints that implement a common interface.

For more information, please refer to the standard described by the W3C.

#### 6.4. NetCDF

NetCDF (network Common Data Form) from Unidata is a set of interfaces for array-oriented data access and a freely distributed collection of data access libraries for C, Fortran, C++, Java, and other languages. The netCDF libraries support a machine-independent format for representing scientific data. Together, the interfaces, libraries, and format support the creation, access, and sharing of scientific data [1].

NetCDF data is:

- Self-Describing. A netCDF file includes information about the data it contains.
- Portable. A netCDF file can be accessed by computers with different ways of storing integers, characters, and floating-point numbers.
- Scalable. A small subset of a large dataset may be accessed efficiently.
- Appendable. Data may be appended to a properly structured netCDF file without copying the dataset or redefining its structure.
- Shareable. One writer and multiple readers may simultaneously access the same netCDF file.
- Archivable. Access to all earlier forms of netCDF data will be supported by current and future versions of the software.

The netCDF software was developed by Glenn Davis, Russ Rew, Ed Hartnett, John Caron, Dennis Heimburger, Steve Emmerson, Harvey Davies, and Ward Fisher at the Unidata Program Center in Boulder, Colorado, with contributions from many other netCDF users.

NetCDF is a binary file format that also contains metadata describing the contents and format of the data in the file. Version 4 of the netcdf library stores data in HDF5 format files; earlier versions stored data in a custom format.

NetCDF files contain one or more variables, which are usually structured as regular N-dimensional arrays. For example, you might have a variable named "Temperature" that is a function of longitude, latitude, and height. NetCDF files also contain dimensions, which describe the extent of the variables' arrays. In our Temperature example, the dimensions are "longitude", "latitude", and "height". Data

can be read from or written to variables in arbitrary hyperslabs (for example, you can read or write all the Temperature values at a given height, or at a given latitude).

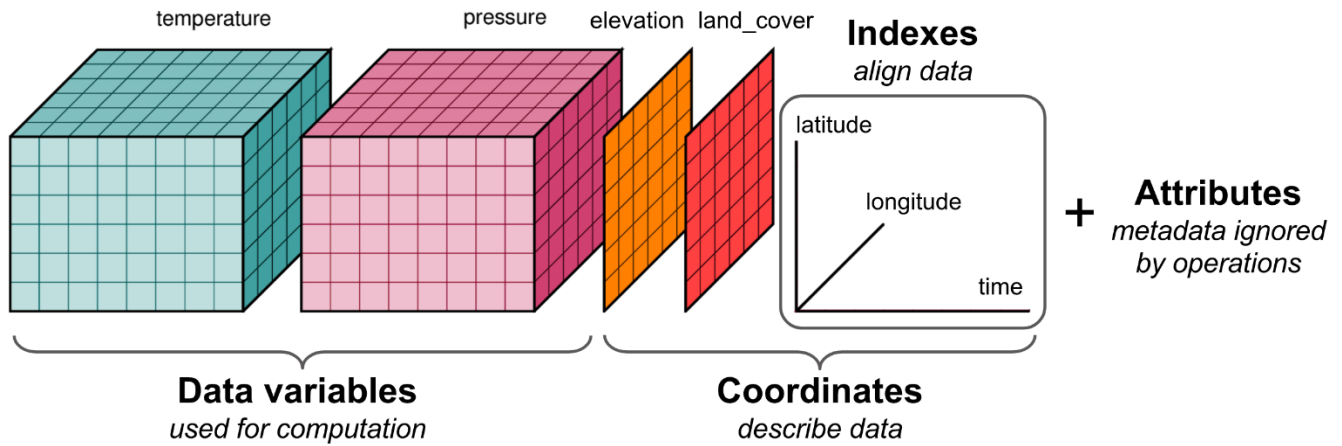


Figure 8: NetCDF structure example

## 7. Standards for Metadata access

Every day, experiments, observations or simulations from scientific projects are generating big amounts of data. Data is created from a variety of disciplines or scientific fields in heterogeneous formats, dimensions and sizes and it is far from being a uniform set of resources.

Metadata, understood as data about data, is an essential element that describes what the data is about, its context and content as well as other important auxiliary information that allows understanding those datasets before analysing the content itself. For example, numeric values of a variable are meaningless if they are not presented with the corresponding units. The use of metadata has two important clients: it must be understood by humans, who need to know more about the data, and it must be accepted by machines, so that it needs to be formally defined and described to make it actionable. Besides, in a big data environment, heterogeneous data coming from different disciplines and sources have to be integrated, and metadata can provide mechanisms to make data interoperable.

The most relevant objectives of Data Management Plans (DMP) include not only details on how data is managed during a Project but also how data could be exploited after the project life and how it could be reused.

This future exploitation needs to be supported by metadata, since in many cases data can be difficult to understand if it is not accompanied by a proper description. It needs to be promoted by academic institutions, universities, and government agencies.

Scientific data have often an initial group of users including mostly specialists that understand the dataset and its context. Their knowledge includes implicit assumptions about the data that allow them to understand the content and they only need initially a limited metadata to work with. In other cases, the metadata does not exist at all and most of the required additional information is in the researcher's head. To avoid this production of "closed" data, and to ensure the potential exploitation beyond the original purpose, metadata need to be created under discipline-oriented or generic standards, so that data can be interpreted by both humans and machines and interoperability can be promoted.

Making metadata useful and precise is not easy since it is essential to promote the processes for data discoverability and reuse, both for humans and machines. There are many problems that researchers face to ensure metadata optimization. First of all, the compatibility among disciplines of metadata standards, which have specific characteristics in terms of a type of data or even units. Secondly, the preservation of datasets and metadata, that sometimes are only supported during the project life due to the limits on funding. Thirdly, the limits to data sharing, for reasons like privacy, legal or security issues. Finally, the level of human dependency, since some datasets or data repositories are managed by specialized professionals.

High-level actors like government agencies or other institutions need to act over those issues in order to design and build a sustainable system that guarantees the implementation of metadata frameworks.

Some examples of metadata standards are:

### **7.1. Dublin Core**

The most basic metadata standard, oriented not only to a dataset but also to describe any digital or even physical object, is Dublin Core.

This metadata standard released in 1995 and composed by fifteen extensible core elements, aims at describing any web-based resource with minimum but complete enough details, but it can be used to define also any digital or physical object. The standard information includes contextual details, like Title, Author, Creator, Description, Date, Type, etc. Thanks to its simplicity, Dublin Core is used by many researchers from different disciplines, including museum curators, and it is the base of other discipline-specific metadata standards. For example, Darwin Core is an extension used in

biodiversity research. The elements included as the core in Dublin Core are also the base for interoperability systems that integrate data resources from different origins

## 7.2. Darwin core (biodiversity)

Darwin Core (often abbreviated to DwC) is an extension of Dublin Core for biodiversity informatics. The Darwin Core is body of standards. It includes a glossary of terms (in other contexts these might be called properties, elements, fields, columns, attributes, or concepts) intended to facilitate the sharing of information about biological diversity by providing reference definitions, examples, and commentaries. The Darwin Core is primarily based on taxa, their occurrence in nature as documented by observations, specimens, samples, and related information. Included are documents describing how these terms are managed, how the set of terms can be extended for new purposes, and how the terms can be used. The normative document for the terms (RDF-NORMATIVE) is written in the Resource Description Framework (RDF) and is the definitive resource to understand the term definitions and their relationships to each other. The Simple Darwin Core (SIMPLEDWC) is a specification for one particular way to use the terms - to share data about taxa and their occurrences in a simply structured way - and is probably what is meant if someone suggests to "format your data according to the Darwin Core".

## 7.3. EML, Ecological Metadata Language (environmental data)

Ecological Metadata Language (EML) is a metadata specification particularly developed for the ecology discipline. It is based on prior work done by the Ecological Society of America and associated efforts (Deliverable 7.3).

Deliverable 7.3 reviews the use of metadata within DANUBIUS-RI. They group metadata in 2 main categories: (i) Descriptive and (ii) Administrative. The first type (Descriptive) is used to identify, authenticate, and describe collections and related trusted information resources whereas the second (Administrative) is dedicated to describe managing and administering data collections and information resources.

The Dublin Core Metadata Initiative (DCMI, 2018) in its most updated version is proposed by Deliverable 7.3 as the vocabulary terms that can be used for Descriptive metadata. Their review summarizes the full list of terms associated to the DCMI.

The administrative metadata is divided into three subcategories following NISO (2018) and based on their common function and scope: (i) Technical metadata (how a system functions, or system-level details about resources), (ii) Preservation metadata (information to support the processes involved in ensuring that a resource continues to exist over time) and (iii) Rights metadata (about



information that may be used to control who gets access to a resource, under what conditions, and what they can do with it). The associated terms are detailed in Amarioarei et al (2017).

Finally, the authors present a complete list of existing standards by domains of knowledge related to metadata.

## 8. Supersites survey

As presented in the ESFRI application and the DANUBIUS-PP Ontology Reference Document (Deliverable 9.1), Supersites are the components of the DANUBIUS-RI distributed Research Infrastructure which will be the test beds of the DANUBIUS-RI scientifically excellent ideas, areas where the developed concepts will be refined and verified.

They will provide natural laboratories for observation, research, modelling and innovation at locations of high scientific importance and opportunity, covering RS systems from river source to transitional waters and coastal seas. Ranging from the near pristine to the heavily impacted, the Supersites will be selected to provide contrasting systems across environmental, social and economic gradients that have been impacted, to varying degrees either directly or indirectly, by industrialisation, urbanisation, population expansion, land use change and farming. They will provide interdisciplinary research platforms and identify, model and define system states and conditions for naturally and anthropogenically triggered transitions in the physical, biogeochemical and biological states. They will provide excellent opportunities to undertake social and economic investigations in contrasting settings.

For collecting the information of every supersite/node/centre, it has been developed a general template. The template not only presents information about the supersite, it also emphasizes standard data format and metadata. It provides some recommendations for the application of the FAIR principles on the respective data.

The template has the following sections:

1. Brief description of supersite/node/centre
2. Description page
3. Interactive pages
4. Data & Metadata
5. Training
6. Publications

All sections are explained in depth in the template. Furthermore, to facilitate the use of the template, it has attached a filled template with the information of the Guadalquivir Estuary Supersite.

A DANUBIUS partner map has been developed too. Clicking on the marker, information about the referred centre/node/supersite is displayed (This information is collected in the first section of the general template).

The map can be accessed in the following link: <https://alnan-gie-us.shinyapps.io/supersites/>

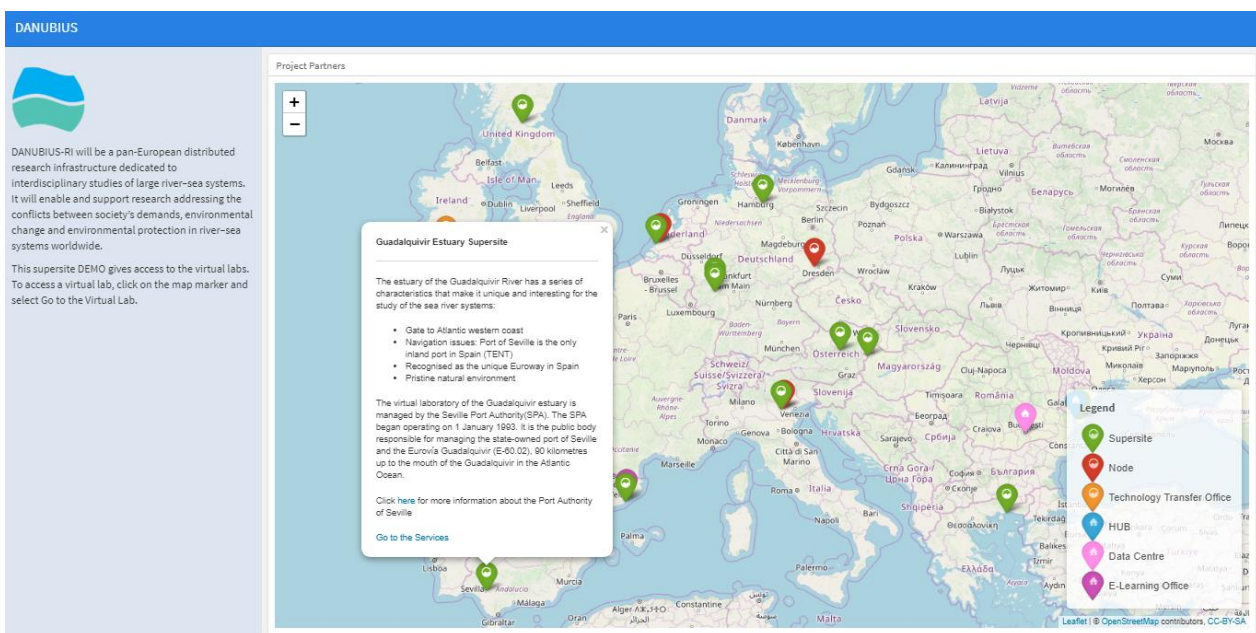


Figure 9: DANUBIUS partners Interactive Map

## 9. Recommendations from D8.2

Environmental and socio-economic science are increasing the volume and complexity of data. Web technologies have allowed complex websites to offer a wide range of e-services permitting the scientific and public in general community to download and use huge amounts of data and metadata.

A wide range of e-services can be found in all kind of disciplines and activities with the aim of offering a set of tools for user specific purposes. The present chapter different initiatives that has been accepted as standards and have developed e-services of interest for DANUBIUS-RI. Among the exiting offer the Open Geospatial Consortium has arisen as one of the most used standards,

### 9.1. The Open Geospatial Consortium (OGC)

According to its web page (OGC, 2018)

“The OGC is an international not for profit organization committed to making quality open standards for the global geospatial community. These standards are made through a consensus process and are freely available for anyone to use to improve sharing of the world's geospatial data”

OGC standards are used in a wide variety of domains including Environment, Defence, Health, Agriculture, Meteorology, Sustainable Development and many more. As it can be seen, OGC is a suitable set of protocols, processes and standards for DANUBIUS.RI.

The Inspire Directive establishes, in relation to services, that the organizations responsible for the geographical information of each of the subjects listed in the annexes of the Directive must offer their information in an integrated and interoperable way through spatial data services. The OGC allow access to geographic information located in the servers of the producer organizations themselves (thus ensuring that it is updated information), in a standard way (the OGC and ISO specifications) and through any application compatible. It pursues agreements between the different companies in the sector that allow the interoperation of their geoprocessing systems and facilitate the exchange of geographic information for the benefit of users.

Table 7 shows the existing standards under the umbrella of OGC. Figure 10 shows the main service and standard components of OGC under a IBM’s web service architecture

The OGC services can be grouped in 4 main categories: (i) Processing services, (ii) Portrayal services, (iii) Data services and (iv) catalogue services.

3dP	KML	Simple Features
ARML2.0	LandInfra/InfraGML	Simple Features CORBA
Cat: ebRIM App Profile: Earth Observation Products	Location Services (OpenLS)	Simple Features OLE/COM
Catalogue Service	Moving Features	Simple Features SQL
CDB	NetCDF	Styled Layer Descriptor
CityGML	Observations and Measurements	Symbology Encoding
Coordinate Transformation	Open GeoSMS	Table Joining Service
Filter Encoding	OpenMI	TimeseriesML (tsml)
GML in JPEG 2000	OpenSearch for EO	WaterML
GeoAPI	OpenSearch Geo	Web Coverage Processing Service
GeoPackage	Ordering Services Framework for Earth Observation Products	Web Coverage Service
GeoSciML	OWS Context	Web Feature Service
GeoSPARQL	PubSub	Web Map Context
Geography Markup Language	PUCK	Web Map Service
GeoRSS	SWE Common Data Model	Web Map Tile Service
Geospatial eXtensible Access Control Markup Language (GeoXACML)	SWE Service Model	Web Processing Service
Geospatial User Feedback (GUF)	Sensor Model Language	Web Service Common
GroundwaterML	Sensor Observation Service	WKT CRS
i3s	Sensor Planning Service	
IndoorGML	SensorThings	

Table 7. List of standards of OGC in alphabetical order. Services are in yellow.

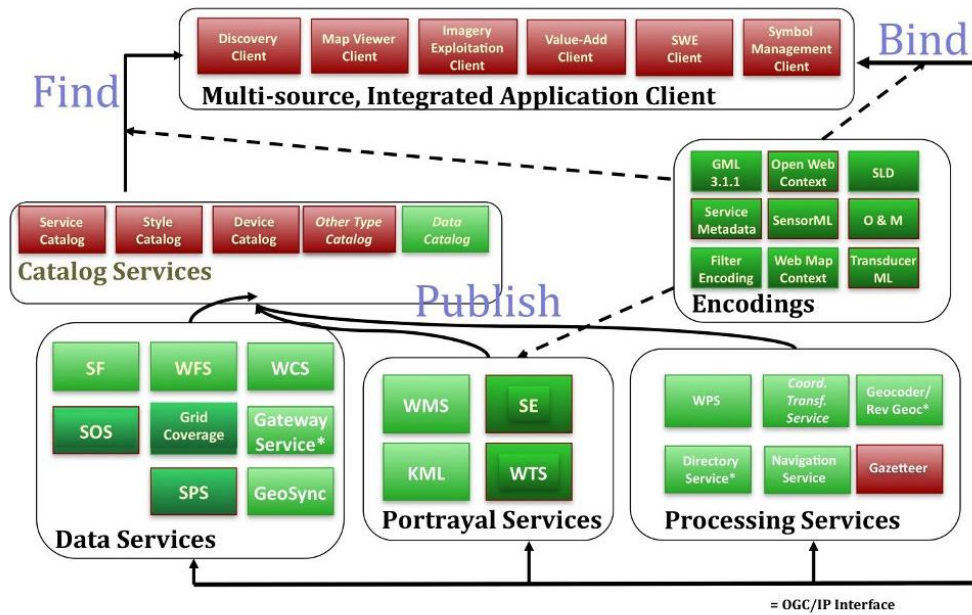


Figure 10. Web services framework of OGC geoprocessing standards (OSGeoLive, 2018)

## 10. Selection of datasets for e-services DEMOnstration

The DEMO shows the capacities of Danubius-PP e-services, a dataset from the Guadalquivir Estuary (generated by University of Málaga) has been chosen. This dataset is composed by 2 simulated models.

The simulated models contain two years (2008 - 2009) of data with a sampling rate of 1 hour. There is a barocline and a bathymetric model. These models are combined in a finite elements mesh in NetCDF format of the Guadalquivir river estuary. The mesh is composed by 7010 nodes, 89 referred to the navigable channel of the river. The navigable channel of the Guadalquivir is 78 km long; between one node and the next there is an average distance of 1km. Each node has 3 geographical properties ("latitude", "longitude" and "deep"). There are 3 variables ("Along River Velocity", "Across River Velocity" and "Salinity") that depend on each node and the "Water Layer". The "Water Layer" refers to the various layers that the depth of the river can be divided into.

The Port of Seville is located inside the estuary of the Guadalquivir, the only inland seaport in Spain. It is in one of the main metropolitan areas of the country, with more than 1.5 million inhabitants in the surrounding area, and in the load-generating centre of the south of the peninsula. In addition, it is a completely multimodal port with an optimum connection between ship, train and truck within the port public domain and has a large area for the establishment of companies.

## 11. Selection of tools for e-services DEMOnstration

The DEMO includes several functionalities: tabs, interactive graphics, maps, ... Below, the tools that have been used to develop it are presented. Some of these tools were also mentioned in the deliverables 8.9 and 8.10.

### 11.1. R

R is a language and environment for statistical computing and graphics. It is a GNU project which is similar to the S language and environment which was developed at Bell Laboratories (formerly AT&T, now Lucent Technologies) by John Chambers and colleagues. R can be considered as a different implementation of S. There are some important differences, but much code written for S runs unaltered under R.



*Figure 11: R logo*

R provides a wide variety of statistical (linear and nonlinear modelling, classical statistical tests, time-series analysis, classification, clustering, ...) and graphical techniques, and is highly extensible. The S language is often the vehicle of choice for research in statistical methodology, and R provides an Open Source route to participation in that activity.

One of R's strengths is the ease with which well-designed publication-quality plots can be produced, including mathematical symbols and formulae where needed. Great care has been taken over the defaults for the minor design choices in graphics, but the user retains full control.

R is available as Free Software under the terms of the Free Software Foundation's GNU General Public License in source code form. It compiles and runs on a wide variety of UNIX platforms and similar systems (including FreeBSD and Linux), Windows and MacOS.

R is an integrated suite of software facilities for data manipulation, calculation and graphical display. It includes

- an effective data handling and storage facility,
- a suite of operators for calculations on arrays, in particular matrices,
- a large, coherent, integrated collection of intermediate tools for data analysis,

- graphical facilities for data analysis and display either on-screen or on hardcopy, and
- a well-developed, simple and effective programming language which includes conditionals, loops, user-defined recursive functions and input and output facilities.

The term “environment” is intended to characterize it as a fully planned and coherent system, rather than an incremental accretion of very specific and inflexible tools, as is frequently the case with other data analysis software.

R, like S, is designed around a true computer language, and it allows users to add additional functionality by defining new functions. Much of the system is itself written in the R dialect of S, which makes it easy for users to follow the algorithmic choices made. For computationally-intensive tasks, C, C++ and Fortran code can be linked and called at run time. Advanced users can write C code to manipulate R objects directly.

Many users think of R as a statistics system. We prefer to think of it as an environment within which statistical techniques are implemented. R can be extended (easily) via packages. There are about eight packages supplied with the R distribution and many more are available through the CRAN family of Internet sites covering a very wide range of modern statistics.

R has its own LaTeX-like documentation format, which is used to supply comprehensive documentation, both on-line in a number of formats and in hardcopy .

## 11.2. Rstudio

RStudio is an integrated development environment (IDE) for R. It includes a console, syntax-highlighting editor that supports direct code execution, as well as tools for plotting, history, debugging and workspace management .

RStudio is available in open source and commercial editions and runs on the desktop (Windows, Mac, and Linux) or in a browser connected to RStudio Server or RStudio Server Pro (Debian/Ubuntu, RedHat/CentOS, and SUSE Linux).

- **RStudio Desktop** is an R IDE that works with the version of R you have installed on your local Windows, Mac OS X, or Linux workstation. RStudio Desktop is a standalone desktop application that in no way requires or connects to RStudio Server.
- **RStudio Server** is a Linux server application that provides a web browser-based interface to the version of R running on the server.

## 11.3. Tidyverse

The tidyverse is an opinionated collection of R packages designed for data science. All packages share an underlying design philosophy, grammar, and data structures.

### 11.3.1. *ggplot2*

Is a plotting package that makes it simple to create complex plots from data in a data frame. It provides a more programmatic interface for specifying what variables to plot, how they are displayed, and general visual properties. Therefore, we only need minimal changes if the underlying data change or if we decide to change from a bar plot to a scatter plot. This helps in creating publication quality plots with minimal amounts of adjustments and tweaking.

*ggplot2* functions like data in the 'long' format, i.e., a column for every dimension, and a row for every observation. Well-structured data will save you lots of time when making figures with *ggplot2*.

*ggplot* graphics are built step by step by adding new elements. Adding layers in this fashion allows for extensive flexibility and customization of plots.

### 11.3.2. *dplyr*

*dplyr* is a package for data manipulation, developed by Hadley Wickham and Romain Francois. It is built to be fast, highly expressive, and open-minded about how your data is stored. It is installed as part of the the tidyverse meta-package and, as a core package, it is among those loaded via `library(tidyverse)`.

*dplyr*'s roots are in an earlier package called *plyr*, which implements the "split-apply-combine" strategy for data analysis (PDF). Where *plyr* covers a diverse set of inputs and outputs (e.g., arrays, data frames, lists), *dplyr* has a laser-like focus on data frames or, in the tidyverse, "tibbles". *dplyr* is a package-level treatment of the `ddply()` function from *plyr*, because "data frame in, data frame out" proved to be so incredibly important.

### 11.3.3. *tidyr*

The goal of *tidyr* is to help you create tidy data. Tidy data is data where:

- Every column is variable.
- Every row is an observation..
- Every cell is a single value.

Tidy data describes a standard way of storing data that is used wherever possible throughout the tidyverse. If you ensure that your data is tidy, you'll spend less time fighting with the tools and more time working on your analysis. Learn more about tidy data in vignette("tidy-data").

#### 11.4. Shiny

Shiny is an R package that makes it easy to build interactive web apps straight from R. You can host standalone apps on a webpage or embed them in R Markdown documents or build dashboards. You can also extend your Shiny apps with CSS themes, htmlwidgets, and JavaScript actions.

Shiny makes it incredibly easy to build interactive web applications with R. Automatic "reactive" binding between inputs and outputs and extensive prebuilt widgets make it possible to build beautiful, responsive, and powerful applications with minimal effort. Shiny applications have two components: a user-interface definition and a server script .

#### 11.5. Flexdashboard

Flexdashboard is an easy way to quickly create dashboards, ideal for publishing visualizations with flexible and easy to specify layouts. You can use flexdashboard to publish groups of related data visualizations as a dashboard. A wide variety of components can be included in flexdashboard layouts, including:

1. Interactive JavaScript data visualizations based on htmlwidgets.
2. R graphical output including base, lattice, and grid graphics.
3. Tabular data (with optional sorting, filtering, and paging).
4. Value boxes for highlighting important summary data.
5. Gauges for displaying values on a meter within a specified range.
6. Text annotations of various kinds.

Flexible options for component layout are available and dashboards with many components can be organized into multiple pages. The flexdashboard layout system also adapts itself intelligently for display on mobile devices.

There are also several ways to add interactivity to flexdashboards. You can use htmlwidgets to add client-side JavaScript interactivity (data tips, panning, zooming, etc.) and Shiny to build fully custom interactions in R .





- Create maps right from the R console or RStudio
- Embed maps in knitr/R Markdown documents and Shiny apps
- Easily render spatial objects from the sp or sf packages, or data frames with latitude/longitude columns
- Use map bounds and mouse events to drive Shiny logic
- Display maps in non spherical mercator projections
- Augment map features using chosen plugins from leaflet plugins repository



Figure 13: Leaflet Screenshot

## 11.7. NetCDF4

Provides a high-level R interface to data files written using Unidata's netCDF library (version 4 or earlier), which are binary data files that are portable across platforms and include metadata information in addition to the data sets.

Using this package, netCDF files (either version 4 or “classic” version 3) can be opened and datasets read in easily. It is also easy to create new netCDF dimensions, variables, and files, in either version 3 or 4 format, and manipulate existing netCDF files. This package replaces the former ncd package, which only worked with netcdf version 3 files. For various reasons the names of the functions have had to be changed from the names in the ncd package.

This package allows reading from, writing to, and creation of netCDF files, either netCDF version 3 or (optionally) netCDF version 4. If you choose to create version 4 output files, be aware that older netcdf software might only be able to read version 3 files.

Note that both the netCDF library and the HDF5 library must already be installed on your machine for this R interface to the library to work.

### 11.8. DT(DataTable)

The R package DT provides an R interface to the JavaScript library DataTables. R data objects (matrices or data frames) can be displayed as tables on HTML pages, and DataTables provides filtering, pagination, sorting, and many other features in the tables.



	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
1	5.1	3.5	1.4	0.2	setosa
2	4.9	3	1.4	0.2	setosa
3	4.7	3.2	1.3	0.2	setosa
4	4.6	3.1	1.5	0.2	setosa
5	5	3.6	1.4	0.2	setosa

Showing 1 to 5 of 150 entries

Previous 1 2 3 4 5 ... 30 Next

Figure 14: DT Screenshot

### 11.9. Rmarkdown

The document format “R Markdown” was first introduced in the knitr package (Xie 2015, 2019c) in early 2012. The idea was to embed code chunks (of R or other languages) in Markdown documents. In fact, knitr supported several authoring languages from the beginning in addition to Markdown, including LaTeX, HTML, AsciiDoc, reStructuredText, and Textile. Looking back over the five years, it seems to be fair to say that Markdown has become the most popular document format, which is what we expected. The simplicity of Markdown clearly stands out among these document formats.

However, the original version of Markdown invented by John Gruber was often found overly simple and not suitable to write highly technical documents. For example, there was no syntax for tables,

footnotes, math expressions, or citations. Fortunately, John MacFarlane created a wonderful package named Pandoc (<http://pandoc.org>) to convert Markdown documents (and many other types of documents) to a large variety of output formats. More importantly, the Markdown syntax was significantly enriched. Now we can write more types of elements with Markdown while still enjoying its simplicity.

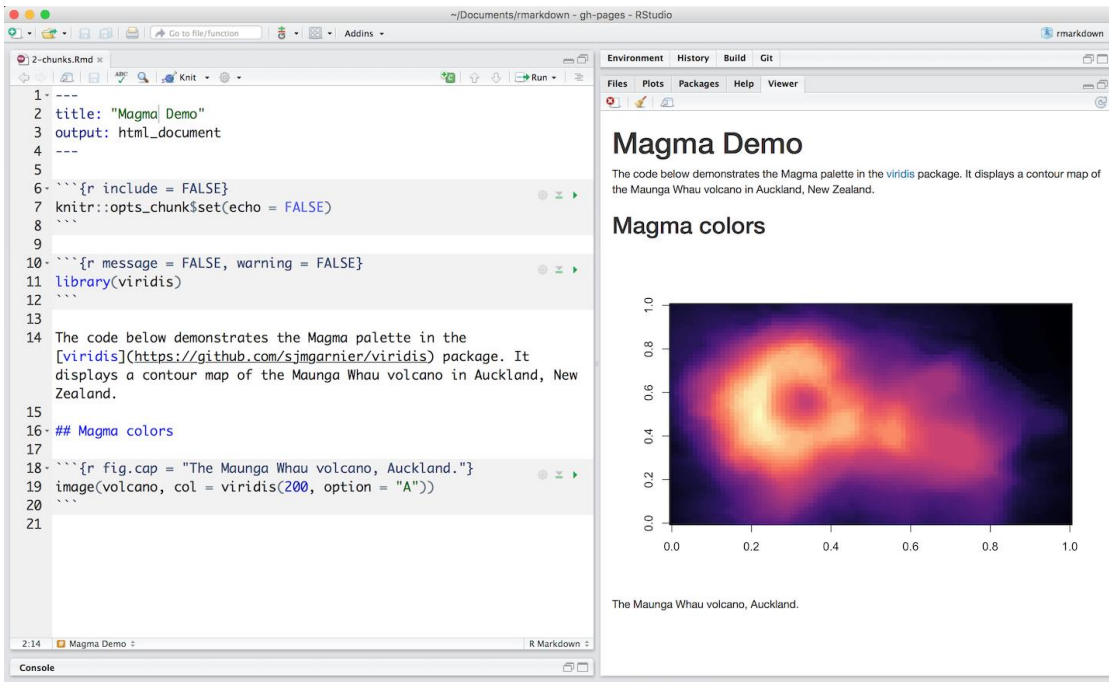


Figure 15: Rmarkdown example Code vs Visualization

In a nutshell, R Markdown stands on the shoulders of knitr and Pandoc. The former executes the computer code embedded in Markdown, and converts R Markdown to Markdown. The latter renders Markdown to the output format you want (such as PDF, HTML, Word, and so on).

The rmarkdown package (J. Allaire, Xie, McPherson, et al. 2019) was first created in early 2014. During the past four years, it has steadily evolved into a relatively complete ecosystem for authoring documents, so it is a good time for us to provide a definitive guide to this ecosystem now. At this point, there are a large number of tasks that you could do with R Markdown:

- Compile a single R Markdown document to a report in different formats, such as PDF, HTML, or Word.
- Create notebooks in which you can directly run code chunks interactively.
- Make slides for presentations (HTML5, LaTeX Beamer, or PowerPoint).

- Produce dashboards with flexible, interactive, and attractive layouts.
- Build interactive applications based on Shiny.
- Write journal articles.
- Author books of multiple chapters.
- Generate websites and blogs.

## 12. User interface DEMO

An interface is a point where users interact with the software they are using. User Interface Design in Information Technology refers to the design of the user interface for software or applications.

A good User Interface is important in the sense that it makes it easier for your target audience to clearly see what your products are. It is designed in a way to display the services that you offer without ambiguity, in order to draw your visitors' attention and keep them on your site.

### 12.1. Requirements

In the data context, visualization is a form of communication that portrays dense and complex information in graphical form. The resulting visuals are designed to make it easy to compare data and use it to tell a story; both of which can help users in decision making. It can express data of varying types and sizes: from a few data points to large multivariate datasets. A good User Interface Design presents a seamless blend of visual design, interaction design, and information architecture:

**Visual Design:** improves the ornamental value of a site through the strategic implementation of elements such as fonts, colors and images, among others.

**Interactive Design:** looks at how users interact with technology. Then it uses the understanding of such interactions to create an interface with well thought-out behaviors. Excellent interactive design not only anticipates how a person interacts with a system, it anticipates and solves problems in time.

**Information Architecture:** is designed to help users find the info they need to complete various tasks. It, therefore, involves labelling, structuring, and organizing the web content in a manner that makes it easily accessible and sustainable.

Everything stems from knowing the users, including understanding their goals, skills, preferences, and tendencies. In the case of an interface oriented to the scientific community, visual design

requirements are always less restrictive. Researchers are accustomed to functional and simple interfaces. For the design of the interface the following have been considered:

- Keep the interface simple and responsive
- Create consistency and use common UI elements.
- Be purposeful in page layout.
- Make sure that the system communicates what is happening.
- Think of defaults values.

### **12.1.1. Responsive**

Responsive Web Design (RWD) is a web design approach that makes web pages render well on a variety of devices and window or screen sizes. Recent work also considers the proximity of the viewer as part of the viewing context as an extension of RWD. Content, design and performance are needed on all devices to ensure usability and satisfaction.

A site designed with RWD adapts the layout to the viewing environment using fluid, proportion-based grids, flexible images, and CSS3 media queries, an extension of the @media rule, in the following ways:

- The fluid grid concept calls for page element sizing to be in relative units like percentages, rather than absolute units like pixels or points.
- Flexible images are also sized in relative units, to prevent them from displaying outside their containing element.
- Media queries allow the page to use different CSS style rules based on characteristics of the device the site is being displayed on, e.g. width of the rendering surface (browser window width or a physical display size).
- Responsive layouts automatically adjust and adapt to any device screen size, whether it is a desktop, a laptop, a tablet, or a mobile phone.

Responsive web design has become more important as the amount of mobile traffic now accounts for more than half of total internet traffic. Therefore, Google announced Mobilegeddon in 2015, and started to boost the ratings of sites that are mobile friendly if the search was made from a mobile device. Responsive web design is an example of user interface plasticity.



Figure 16: Responsive Design

### 12.1.2. Visual/Interactive Design and Information Structure

An important feature from the point of view of GUI functionality is that the end user is able to use it intuitively. In this case the end user are mainly researchers and scientists who have knowledge in the field of study concerned. Due to the multiple dimensions of the attributes, to complete information, have been used color scales, click popups, hover popups, etc ...

The DEMO interface contains several maps that allow you to navigate through them and click on each node of the mesh to obtain physical information about it: latitude, longitude and depth. Charts that superimpose several levels or sections over the same time interval allow the user to activate or deactivate the different levels to facilitate the visualization. They also allow the user to obtain information by passing the mouse over the different lines. The tables permit to sort by a predetermined column, download in different formats and print. For all pages (with the exception of the description page) there are filters for date and time, date range, levels, sections, ...

All pages have a description on the left side that explains what data is being displayed and how to interact with each of them. It has been considered that as its main use is going to be the research work, short incisive descriptions are enough.

## 12.2. Overview

As said before, this DEMO interface has been developed using R. The interface is a dashboard where users can get some information of the referred dataset (Guadalquivir Estuary dataset). The interface is composed by eight pages. In these pages, descriptions, graphics, tables, maps, metadata and data info, publications, etc., are shown.

### 12.2.1. Layout

In this section is presented the different elements that make up the Layout of the developed graphical interface. All these elements are typical of Flexdashboard, a Shiny based R library for dashboarding. This library is explained in section 11.4.

- Column: Basic elements that compose the scheme of each of the pages of the platform
- Sidebar: Not shown on all pages. In the sidebar there are filters that allow to represent in 2D relations between the magnitudes of our model, keeping some fixed dimensions.
- Graphs: The vast majority in 2D and colour, represent the different magnitudes measured. These graphs are made with ggplot2
- Tables: Dynamic tables made with the DT package, to visualize the selected points and also to download the information.
- Maps: Dynamic maps developed with the Leaflet package. They allow to navigate and visualize the positioning of the nodes. In addition, they also contain popups that show information about positioning and depth.



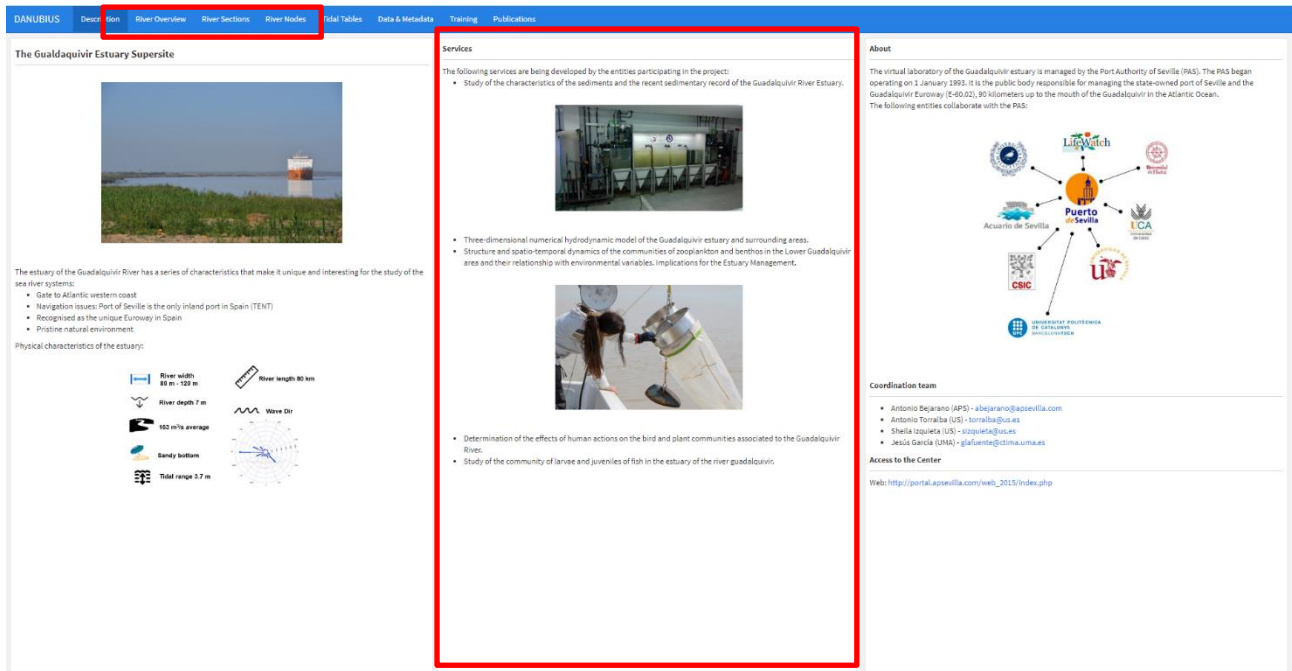


Figure 17: Layout Image 1

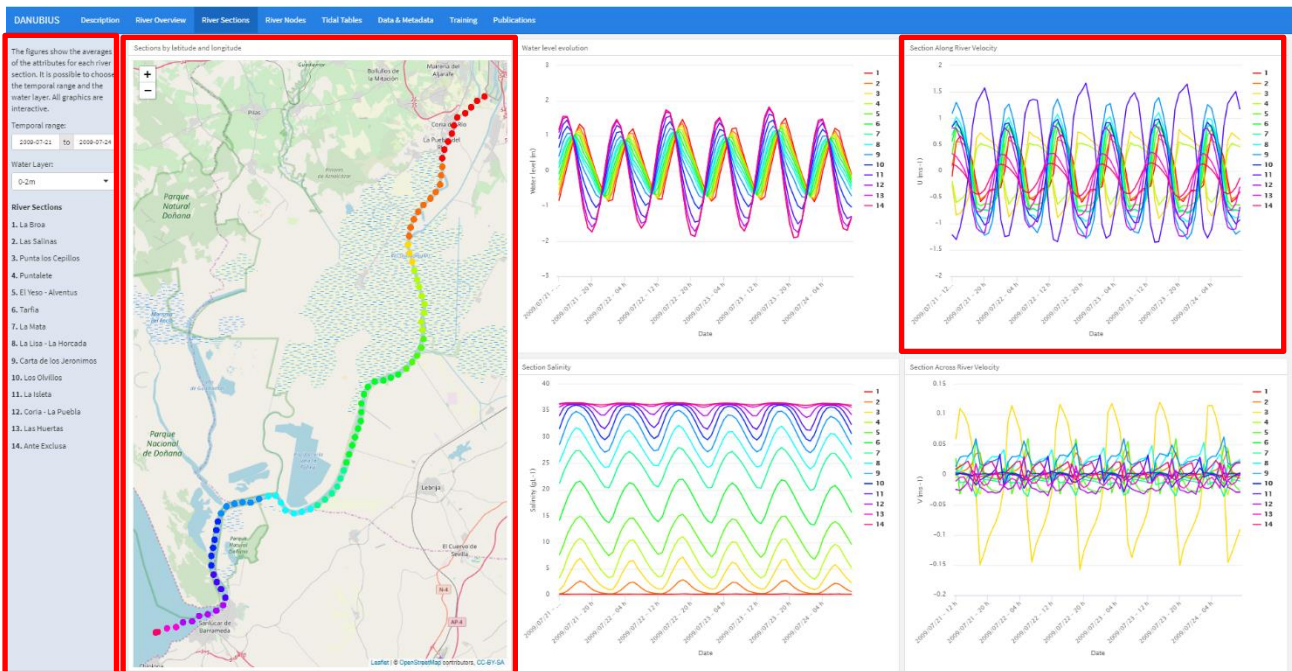


Figure 18: Layout Image 2

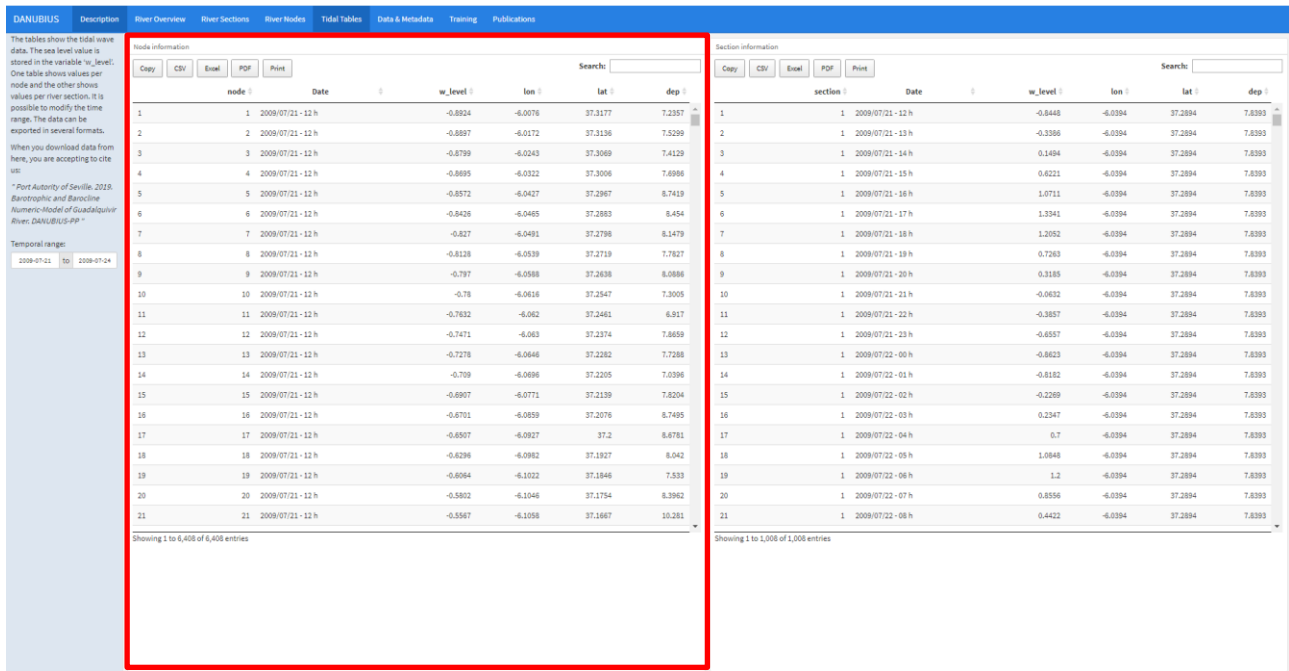


Figure 19: Layout Image 3

## 12.2.2. Pages

All DEMO pages are described briefly here. All the information represented in the interface has been collected using the general template explained in section 8 of this document.

### 12.2.2.1. Description

This is the first section of the interface. It shows all the information about the research centre, some images, links, managers, etc. In this case, it is composed of three independent columns. This page takes the following information from the template: Introduction, About, Services, Coordination team, Access to the center, eTraining, Publications.

### 12.2.2.2. River Overview

This second page of the interface contains 6 graphics and a map. The 6 graphics follow the silhouette of the Guadalquivir river. As it can be seen in the map header, these visualizations are made up of 89 nodes. In the sidebar part there are 3 filters: date, time and water sheet. The graphs show each attribute in a colour scale. In the main map, the nodes are shown grouped, as the zoom gets closer, the closest nodes are shown.

### 12.2.2.3. River Sections

This interface page shows the Guadalquivir river by sections. These 14 sections are generally used to divide the Guadalquivir; their names are shown on the sidebar. The layout of the page consists of

4 graphs and a map. The graphs show one attribute per time interval; the different colours refer to the different sections. The graphs are interactive, besides obtaining information by passing the mouse over them, it is also possible to remove sections to facilitate visualization. In the sidebar, in addition to the description and the names of the sections, there are two filters: date interval and water layer. The figures show velocities, tidal wave and salinity.

#### 12.2.2.4. River Nodes

This interface page shows 4 graphs and a map. It shows graphs of velocities (along and across river), tidal wave and salinity. Each of these graphs shows the evolution of these variables over a given time range. The values displayed are referred to a singular node. In addition, there is a color scale representing each of the water layers. The only filter used on this page is the time range.

#### 12.2.2.5. Tidal Tables

This page shows two tables that give information about the tidal wave. One shows averages by section and the other shows individual values for each node. Both tables can be downloaded in various formats and printed. The unique filter is the data range.

#### 12.2.2.6. Data & Metadata

In this tab, the data, its structure, and associated metadata are explained. Therefore, in this section, those aspects related to the data type (standards), or the level of accomplishment with FAIR principles must be detailed. Here are also listed the terms of use, license and cite us.

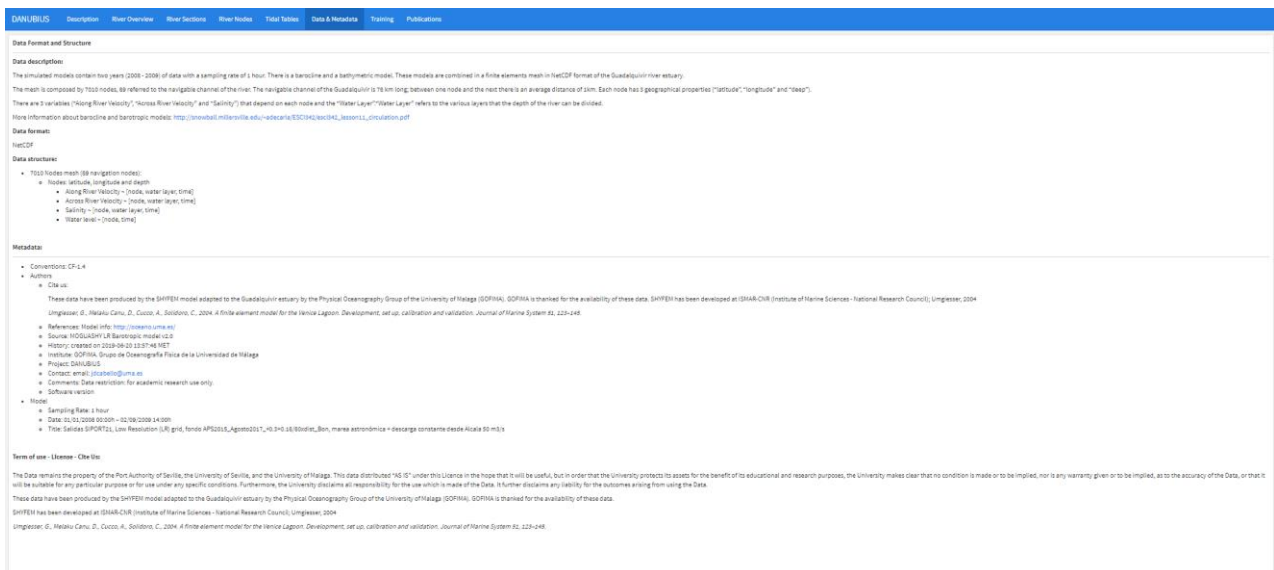


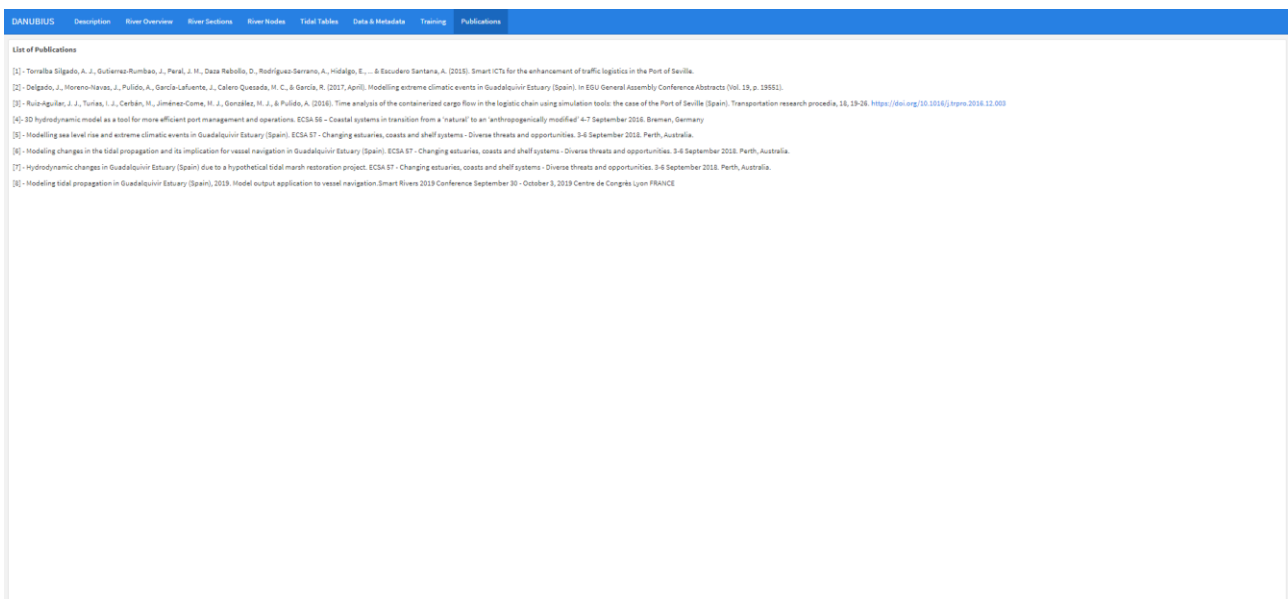
Figure 20: DEMO Data & Metadata

### 12.2.2.7. Training

This page should show a list of training courses dedicated to the centre's services. Guadalquivir Estuary does not have any training material at the moment, so this page is empty.

### 12.2.2.8. Publications

List of 8 publications related to the services offered by Guadalquivir Estuary:



The screenshot shows a web interface for the DANUBIUS project. At the top, there is a navigation bar with tabs for Description, River Overview, River Sections, River Nodes, Tidal Tables, Data & Metadata, Training, and Publications. The Publications tab is selected. Below the navigation bar, the page title is "List of Publications". The main content area contains a list of 8 publications, each with a numbered reference and a brief description of the work.

Ref.	Publication Title
[1]	Torralla Silgado, A. J., Gutiérrez-Rumbao, J., Paraj, J. M., Daza Rebollo, D., Rodríguez-Serrano, A., Hidalgo, E., ... & Escudero Santana, A. (2018). Smart ICTs for the enhancement of traffic logistics in the Port of Seville.
[2]	Delgado, J., Moreno-Navas, J., Pulido, A., García-Lafuente, J., Calero Quesada, M. C., & García, R. (2017, April). Modelling extreme climatic events in Guadalquivir Estuary (Spain). In EGU General Assembly Conference Abstracts (Vol. 19, p. 19551).
[3]	Ruiz-Aguilar, J. J., Turiel, I. J., Cerbán, M., Jiménez-Come, M. J., González, M. J., & Pulido, A. (2016). Time analysis of the containerized cargo flow in the logistic chain using simulation tools: the case of the Port of Seville (Spain). Transportation research procedia, 18, 19-26. <a href="https://doi.org/10.1016/j.trpro.2016.12.003">https://doi.org/10.1016/j.trpro.2016.12.003</a>
[4]	3D hydrodynamic model as a tool for more efficient port management and operations. ECSA 58 - Coastal systems in transition from a 'natural' to an 'anthropogenically modified' 4-7 September 2018. Bremen, Germany
[5]	Modeling sea level rise and extreme climatic events in Guadalquivir Estuary (Spain). ECSA 57 - Changing estuaries, coasts and shelf systems - Diverse threats and opportunities. 3-6 September 2018. Perth, Australia.
[6]	Modeling changes in the tidal propagation and its implication for vessel navigation in Guadalquivir Estuary (Spain). ECSA 57 - Changing estuaries, coasts and shelf systems - Diverse threats and opportunities. 3-6 September 2018. Perth, Australia.
[7]	Hydrodynamic changes in Guadalquivir Estuary (Spain) due to a hypothetical tidal marsh restoration project. ECSA 57 - Changing estuaries, coasts and shelf systems - Diverse threats and opportunities. 3-6 September 2018. Perth, Australia.
[8]	Modeling tidal propagation in Guadalquivir Estuary (Spain). Model output application to vessel navigation Smart Rivers 2019 Conference September 30 - October 3, 2019 Centre de Congrès Lyon FRANCE

Figure 21: DEMO list of Publications

The full interface can be seen at the following link: [https://alnan-gie-us.shinyapps.io/flexdashboard\\_merged\\_v5/](https://alnan-gie-us.shinyapps.io/flexdashboard_merged_v5/)

## 13. Conclusions

This document summarizes all the information and requirements necessary for developing an application interface. The development of an interface must follow principles of design and functionality, in addition to a series of standards.

Some of the most remarkable standards related to Web services, authentication, data and metadata have been presented in this deliverable. Standards facilitate the work of integration and coordination of services, whereas metadata structure is intrinsically related to the field of research to which they belong and to the type of data to which they refer. In addition, the interface design must be visual, interactive and with a proper information structure. All these points support the need of an integration template that serve as a guide for all service providers that form part of the DANUBIUS community.

In this regard, and in order to show the full potential of data interfaces and the usefulness of having a common guide document for all members (the service integration template), a DEMO interface has been created. This DEMO, developed with R, contains the information of one of the supersites that are part of the DANUBIUS community (The Guadalquivir Estuary Supersite. Port Authority of Sevilla, Spain). Using the same tool, a map of the DANUBIUS partners has been created as well.

<https://alinan-gie-us.shinyapps.io/supersites/>

[https://alinan-gie-us.shinyapps.io/flexdashboard\\_merged\\_v5/](https://alinan-gie-us.shinyapps.io/flexdashboard_merged_v5/)

## 14. References

- [1] Wikipedia Metadata Standards. ([https://en.wikipedia.org/wiki/Metadata\\_standard](https://en.wikipedia.org/wiki/Metadata_standard) )
- [2] GO-FAIR principles, (<https://www.go-fair.org/fair-principles/>)
- [3] Architectural Styles and the Design of Network-based Software Architectures, (<http://www.ics.uci.edu/~fielding/pubs/dissertation/top.htm> )
- [4] Second Generation Web Services, (<http://www.xml.com/pub/a/2002/02/06/rest.html> )
- [5] Rest Wiki, (<http://internet.conveyor.com/RESTwiki/moin.cgi> )
- [6] Representational State Transfer, (<http://www.xfront.com/REST.ppt> )
- [7] Costello, Roger L. REST and the Real World, (<http://www.xml.com/lpt/a/2002/02/20/rest.html> )
- [8] Prescod, Paul The Emperor's New Tags - The SOAP/REST Controversy, ([http://www.prescod.net/rest/soap\\_rest\\_short.ppt](http://www.prescod.net/rest/soap_rest_short.ppt) )
- [9] Prescod, Paul A New Direction for Web Services, (<http://www.sys-con.com/xml/article.cfm?id=454> )
- [10] Costello, Roger L. Hypertext Transfer Protocol -HTTP/1.1, (<http://www.ietf.org/rfc/rfc2616.txt> )
- [11] Rest Services, (<https://www.predic8.com/rest-webservices.html> )
- [12] <https://www.chakray.com/en/rest-web-servicies-y-apis-diseno-restful/>
- [13] Overview of SAML, (<https://developers.onelogin.com/saml> )
- [14] What is eduGAIN, (<https://edugain.org/about-edugain/what-is-edugain/> )
- [15] Darwin Core, (<http://rs.tdwg.org/dwc/> )
- [16] WSDL Tutorial, (<https://searchmicroservices.techtarget.com/tutorial/WSDL-Tutorial> )
- [17] JSON – Introduction, ([https://www.w3schools.com/js/js\\_json\\_intro.asp](https://www.w3schools.com/js/js_json_intro.asp) )
- [18] EML - Ecological Metadata Language, (<http://www.dcc.ac.uk/resources/metadata-standards/eml-ecological-metadata-language> )
- [19] Brils J. et al. (2017) Ontology for advanced studies on river-sea systems delivered through a distributed research infrastructure. . Deliverable 9.1, DANUBIUS-PP EU project (GA 739562). Internal report.
- [20] Welcome to The Open Geospatial Consortium, (<https://www.opengeospatial.org/> )
- [21] Amarioarei A., Paun M., Paun A., Sidoroff M., Tusa I., Paraschiv M., Groom S., Martinez V. and Dietrich S. (2017) Metadata description and formalizing documentation procedures. Deliverable 7.3, DANUBIUS-PP EU project (GA 739562). Internal report.

- [22] Gracia V. and Sánchez-Arcilla A. (2017) Review of Research Infrastructure service offerings relevant to DANUBIUS-RI. Deliverable 8.1, DANUBIUS-PP EU project (GA 739562). Internal report.
- [23] Gracia V. (2018) Definition of and recommendations on DANUBIUS-RI remote service offering. Deliverable 8.2, DANUBIUS-PP EU project (GA 739562). Internal report.
- [24] User Interface Design Basics, (<https://www.usability.gov/what-and-why/user-interface-design.html> )
- [25] What is UI Design and why is it important, (<https://f5-studio.com/articles/what-is-user-interface-design-and-why-is-it-important/> )
- [26] Umgiesser, G., Melaku Canu, D., Cucco, A., Solidoro, C., 2004. A finite element model for the Venice Lagoon. Development, set up, calibration and validation. Journal of Marine System 51, 123–145.
- [27] NISO, (<https://www.niso.org/> )
- [28] Rstudio home page, (<https://rstudio.com/> )
- [29] Tidyverse - R packages for data Science, (<https://www.tidyverse.org/> )
- [30] Shiny- Interact. Analyze. Communicate, (<https://shiny.rstudio.com/> )
- [31] Flexdashboard: Easy interactive dashboards for R, (<https://rmarkdown.rstudio.com/flexdashboard/> )
- [32] Open-source JavaScript library for mobile-friendly interactive maps, (<https://leafletjs.com/> )
- [33] Package 'ncdf4', (<https://cran.r-project.org/web/packages/ncdf4/ncdf4.pdf> )
- [34] An R interface to the DataTables library, (<https://rstudio.github.io/DT/> )
- [35] R Markdown. Analyze. Share. Reproduce, (<https://rmarkdown.rstudio.com/> )

# 15. GENERAL TEMPLATE

## DANUBIUS PARTNERS

### 1. BRIEF DESCRIPTION OF SUPERSITE/NODE/CENTRE

A brief summary that describes the research centre, its activities and the services it offers.

Link to the institutional website.

Is it a node / supersite within the project?

All this information will be added to an interactive map that will send the user to the respective DANUBIUS services (max 800 characters including spaces).

### 2. DESCRIPTION PAGE

This is the first page of the interface, which will be composed of several columns depending on the length of its content. In this page the supersite / node / hub, services, coordination team, etc., are presented.

As the home page, it is recommended to add some pictures for the different sections.

#### *Introduction:*

Brief introduction with the main information of the supersite / node / hub.

(max 590 characters including spaces. Optional, but recommended, photo 300 dpi)

#### *About:*

Technical information of the supersite / node / hub and collaborations with other entities.

(max 1.700 characters including spaces. Optional, but recommended, photo 300 dpi)

#### *Services:*

(max 2.700 characters including spaces. Optional, but recommended, photo 300 dpi)



### **Coordination team:**

Members responsible for the service, including Name and Surname, Affiliation, e-mail and optional, but recommended photo 300 dpi

### **Access to the Centre:**

Link to the Centre

## **3. INTERACTIVE PAGES**

From these tabs users will access the services offered by the DANUBIUS' partner such as virtual labs, data repositories, or data models.

A virtual lab is an interactive environment for creating and conducting simulated experiments: a playground for experimentation.

A data repository is a large database infrastructure that collect, manage, and store data sets for data analysis, sharing and reporting.

A data model is an abstract model that organizes elements of data and standardizes how they relate one to another and to the properties of real-world entities).

\*It is possible to add all the page that are necessary

\*\* It is possible to add all the column that are necessary

\*\*\* It is possible to choose and add all the figures, maps, tables, text or filters that are necessary.

The following layout is an example of a virtual lab structured by columns. In this case, Column 0 is the sidebar. Sidebar is composed by text and filters.

### **PAGE X\* – NAME OF THE PAGE**

#### **LAYOUT:**

- Column 0 - Left sidebar\*\*:
  - Text: \*\*\*
  - Filter: \*\*\*
- Column x\*\*
  - Map\*\*\*
  - Figure\*\*\*
  - Table\*\*\*
- Column xx\*\*
  - Map\*\*\*
  - Figure\*\*\*
  - Table\*\*\*

## PAGE Y\* – NAME OF THE PAGE

### LAYOUT:

- Column 0 - Left sidebar\*\*:
  - Text\*\*\*
  - Filter\*\*\*
- Column x\*\*
  - Map\*\*\*
  - Figure\*\*\*
  - Table\*\*\*
- Column xx\*\*
  - Map\*\*\*
  - Figure\*\*\*
  - Table\*\*\*

## 4. DATA and METADATA

In this tab, the data, its structure, and associated metadata are explained. Therefore, in this section, those aspects related to the data type (standards), or the level of accomplishment with FAIR principles must be detailed.

Regarding the type of data that will be available/produced within DANUBIUS-RI, the deliverable 7.3, DANUBIUS-PP EU project (GA 739562). Internal report defines 4 main categories: (i) Observational data, (ii) Experimental data, (iii) Computational outputs and (iv) Compiled data (produced by applying a transformation process on existing data). In parallel, data can be qualitative or quantitative. The data structures identified by the authors are: (i) raster, (ii) vector data, (iii) gridded or array-oriented data, (iv) hierarchical data and (v) data time series. Table with standard formats in **ANNEX 1**.

For example, netCDF is the standard format used by the University of Málaga to gather the outputs of the barocline and barotropic models for the Guadalquivir estuary (Guadalquivir Interface DEMO).

Moreover, as abovementioned, the correspondence of data and metadata with FAIR principles should be also highlighted in this tab (FAIR - Findable, Accessible, Interoperable and Reusable. See **ANNEX 2**.

### *Data Format and Structure*

#### Data description:

Brief explanation of the data. variables, sampling rate, location, ...

#### Data format:

E.g. If the data are from a model, NetCDF format; if the data are historical values of variables, csv format; etc.

#### Data structure:

Explanation (schema) of how data are structured and / or the relationships between them; also dimensions and units



### **Metadata**

*This section contains the information about metadata and its structure. Metadata must follow certain standards in addition to complying with FAIR principles. **ANNEX 3** explains what metadata are, how to use the standards and how to apply the FAIR principles; as well as an example of application on the metadata of the Guadalquivir Estuary Supersite.*

### **Term of use - License - Cite Us**

For sharing the data is mandatory to have a "cite as", a "terms of use" and a License. Some examples are shown in **ANNEX 4**.

## **5. eTRAINING**

List of training courses dedicated to the service, if any.

## **6. PUBLICATIONS**

List of all publications related to the service or services offered by the supersite / node / hub

## ANNEX 1: DATA STANDARDS

The recommended data formats can be seen in the following table:

FORMATS	
Qualitative data	.XML (text according to an appropriate document type) .RTF (rich text format), .TXT (plain text data), ASCII .doc/.docs( widely –used proprietary formats, MS Word), .tex (LaTeX )
Quantitative tabular data with extensive metadata	Data sets with variable labels Data and metadata inside a relational database .XML file with metadata (WaterML2 etc)
Quantitative tabular data with minimal metadata	.CSV (comma-separated values),tab (tab-delimited file) Widely used formats MS Excel (.xls/.xlsx), MS Access (.mdb/.accdb), open documents spreadsheet (.ods)
Quantitative gridded/array-oriented data, self-describing formats	.nc (netCDF) with CF conventions .grb (GRIB) .hdf
Geospatial data	(vector and raster data), tabular GIS attribute data OGC simple features, proprietary ESRI-shp, ...
Digital audio data	.wav (Waveform Audio Format) .mp3 (MPEG audio format)
Digital video data	.MP4 (MPEG 4) .jp2 (motion JPEG)
Documentation and scripts	Files in the .RTF, .PDF or .HTM format, plain text or widely used proprietary formats MSWORD(.doc/.docx), MS EXCEL(.xls/.xlsx), SQL, JAVA, JAVASCRIPT

## ANNEX 2: FAIR PRINCIPLES

As commented in <https://www.go-fair.org/fair-principles/>; in 2016, the 'FAIR Guiding Principles for scientific data management and stewardship' were published in Scientific Data. The authors intended to provide guidelines to improve the findability, accessibility, interoperability, and reuse of digital assets. The principles emphasise machine-actionability (i.e., the capacity of computational systems to find, access, interoperate, and reuse data with none or minimal human intervention) because humans increasingly rely on computational support to deal with data as a result of the increase in volume, complexity, and creation speed of data.

**Findable:** The first step in (re)using data is to find them. Metadata and data should be easy to find for both humans and computers. Machine-readable metadata are essential for automatic discovery of datasets and services, so this is an essential component of the FAIRification process.

- F1. (Meta)data are assigned a globally unique and persistent identifier
- F2. Data are described with rich metadata (defined by R1 below)
- F3. Metadata clearly and explicitly include the identifier of the data they describe
- F4. (Meta)data are registered or indexed in a searchable resource

**Accessible:** Once the user finds the required data, she/he needs to know how can they be accessed, possibly including authentication and authorisation.

- A1. (Meta)data are retrievable by their identifier using a standardised communications protocol
  - A1.1 The protocol is open, free, and universally implementable
  - A1.2 The protocol allows for an authentication and authorisation procedure, where necessary
- A2. Metadata are accessible, even when the data are no longer available

**Interoperable:** The data usually need to be integrated with other data. In addition, the data need to interoperate with applications or workflows for analysis, storage, and processing.

- I1. (Meta)data use a formal, accessible, shared, and broadly applicable language for knowledge representation.
- I2. (Meta)data use vocabularies that follow FAIR principles
- I3. (Meta)data include qualified references to other (meta)data

**Reusable:** The ultimate goal of FAIR is to optimise the reuse of data. To achieve this, metadata and data should be well-described so that they can be replicated and/or combined in different settings.

- R1. Meta(data) are richly described with a plurality of accurate and relevant attributes
  - R1.1. (Meta)data are released with a clear and accessible data usage license
  - R1.2. (Meta)data are associated with detailed provenance
  - R1.3. (Meta)data meet domain-relevant community standards

## ANNEX 3: METADATA, STRUCTURES AND EXAMPLES

The main difference between data and metadata is mode of use. Metadata is not just for data, it is also for users, software services, computing resources. Metadata is not just for description and discovery; it is also for contextualisation (relevance, quality, restrictions (rights, costs)) and for coupling users, software and computing resources to data (to provide a Virtual Research Environment). Metadata must be machine understandable as well as human understandable. Below some examples of typical metadata fields are shown:

- **User metadata:** user identifier; password; digital signature; organisation and possibly further metadata required for disambiguation and identification. Authentication is done by using associated rights (responsibilities and authorities) metadata.
- **Discovery metadata:** dataset, software or resource: identifier; name; abstract; keywords; characteristics such as size, type; rights information.
- **Contextual metadata:** used for explanation of results if required. Such as person; organisation; project; related outputs (publications, products), datasets, software, computing resources available

The structure of the metadata is intrinsically related to the field of research to which they belong and to the type of data to which they refer. There are infinite standards to define metadata. Some examples of metadata structures related to biology and the study of water are shown below:

### 1. Metadata specific to surface underway data

- Underway collection details
  - Ship and cruise identifier.
  - Project (if applicable).
  - Date/time of the start/end of the sampling.
  - Location of the all individual sensors (Water inlet and meteorological).
  - Method of positional fixing (i.e. GPS, DGPS).
  - Depth of the water intake.
  - Flow volume per unit time, flow rate, size of water line.
  - Time lag for water moving between sensors (if applicable).
  - Any other pertinent information (i.e. shading of PAR sensor)
- Instrument details
  - Instrument description, reference number, manufacturer, model, principle of measurement, method of recording — provide a literature reference, web site reference or briefly describe. Include — accuracy, resolution and response range of individual sensors.
  - Instrument modifications and their effect on the data.
- Sample data — for calibration purposes
  - Details of water sampling system (automatic — manufacturer, model, number serial number or manual).
  - Water sampling method, volume and time interval.
  - Details of analyses and the elapsed time between collection and analysis.

- Corrections and calibrations
  - Correction for time lag between sensors.
  - Correction for ships movement — for wind speed and direction.
  - Report on any other corrections made (i.e. correction for increase in water temperature due to flow through ships intake lines).
  - Laboratory calibrations.
  - In-situ calibrations (i.e. comparisons with lowered near surface CTD or water samples).
  - Equations used for computed values (excluding simple averaging).
  - Adjustments made due to variations in calibrations.
- Data sampling/processing
  - Filtering, de-spiking or smoothing methods.
  - Editing and quality control methods.
  - Quality control report.

## 2. Metadata specific to water sample data

- Collection details
  - Ship and cruise identifier.
  - Project (if applicable).
  - CTD station identifier (if appropriate).
  - Sample identifier (i.e. bottle number, rosette position or firing sequence).
  - Date/time of the measurement.
  - Position and method of position fix (i.e. DGPS, GPS).
  - Water column depth (if applicable) — include method — i.e. sounding and methodology, chart.
  - Depth of sample (if applicable).
- Instrument details
  - Instrument description, reference number, manufacturer and model, sampling rate (as appropriate) — provide a literature reference, web site reference or briefly describe.
- Analysis
  - A description of or reference to full laboratory methods and procedures.
  - Details of any external sample analysis, including the laboratory name and accreditation level.
  - A description of or reference to any internal or external quality assurance procedures.
- Data sampling/processing
  - Editing and quality control methods.
  - Identification of trace values (i.e. values below the detection limit).
  - Handling of missing values (null versus zero, or 'blanks').
  - Equations used for computed values.
  - Precision of methods (i.e. number of significant figures).
  - Quality control report.

- Units per volume and per mass

### 3. Metadata specific to biological data

- Collection details
  - Ship and cruise identifier.
  - Project (if applicable).
  - Net tow (or bottle) identifier.
  - Date/time of the start/end of each net tow.
  - Position of the start/end of each net tow.
  - Tow orientation (vertical, horizontal or oblique).
  - Maximum and minimum depth of the net tow.
  - Length and speed of tow.
  - Volume of water filtered.
  - Date, time and depth of bottle sample.
  - Position of bottle sample.
  - Method of position fixing (i.e. DGPS, GPS).
  - Sample preservation (i.e. pickled or frozen).
  - Water column depth. Include method — i.e. sounding and methodology, chart.
  - Weather conditions — including sun and wind.
  - Twilight time.
- Equipment details
  - Net type (i.e. Single or multi-net, net mouth size, net mesh size)
  - Bottle sampling. Include — manufacturer, reference number, rosette position or firing sequence. Provide a literature reference, web site reference or briefly describe.
- Analysis, experiments and data processing
  - Laboratory procedures and instrumentation.
  - Species counts or mass measurements.
  - Description of any respiration, feeding or physiological experiments and results.
  - Any sub-sampling.
  - Quality control methods.
  - Quality control report.

More metadata's structures here:

[https://www.bodc.ac.uk/submit\\_data/submission\\_guidelines/](https://www.bodc.ac.uk/submit_data/submission_guidelines/)

As commented, there are infinite fields that can appear in the metadata, one that increases richness, are the **keywords**. Keywords help you to classify your content. Keywords define a classification scheme (visualized as a taxonomy tree) that provides a knowledge map of your data. Applying these keywords to metadata fields or using as a metadata field allows you to define relationships between content.



The structure of the metadata, as already mentioned, is closely related to the data format and the study area. In order to show a more significant example, we are going to present a data structure associated with the data used for the development of the DEMO interface of the Guadalquivir Estuary.

The data used in the DANUBIUS-pp Guadalquivir Estuary DEMO is in netCDF format. NetCDF metadata follows the CF (Climate and Forecast) standard. CF provides guidelines and recommendations on how to place usage metadata within the netCDF file (<http://cfconventions.org/>). As climate model data are often not simply representative of points in time and space, the standard can provide for the description of coordinate intervals, multidimensional cells and climatological time coordinates. Each file consists of two sections:

**CF Header.** The CF metadata header is split into several components, namely; dimensions, variables and global attribute:

- Dimensions
- Variables
  - Each listed item describes the following information
    - Data type
    - Units
    - Content description
    - Missing data values
    - Maximum and minimum values
- Global Attributes
  - Conventions
  - title
  - insititute
  - source
  - history
  - references
  - comment
  - author

**The data section.** This section of the file contains the actual values for each dimension and variable listed in the order specified in the CF header.

Finally, taking into account the nature of the data, the type, the FAIR principles and some other considerations, a structure has been designed for the metadata of the Guadalquivir estuary.

- Dimensions
- Variables
  - Each listed item describes the following information
    - Data type
    - Units
    - Content description
    - Missing data values

- Global Attributes
  - Conventions
  
  - Author
    - Author/s name:
    - License
    - Terms of Use
    - Cite us
    - References
    - Source
    - History
    - Institute
    - Project
    - Contact
    - Comments
    - Software version
  
  - Model
    - Sampling Rate
    - Date
    - Title
    - Global Unique identifier (e.g.: DOI, PID, ...)
    - Metadata link
  
  - Keywords

## ANNEX 4: TERMS OF USE, LICENSES AND CITE US EXAMPLES

### License (Option 1): Free template

*The Data remains the property of **the Institution**.*

*This data distributed "AS IS" under this License in the hope that it will be useful, but in order that the **Institution** protects its assets for the benefit of its educational and research purposes, the **Institution** makes clear that no condition is made or to be implied, nor is any warranty given or to be implied, as to the accuracy of the Data, or that it will be suitable for any particular purpose or for use under any specific conditions. Furthermore, the **Institution** disclaims all responsibility for the use which is made of the Data. It further disclaims any liability for the outcomes arising from using the Data.*

### License (Option 2): Creative Commons License generator

Creative Commons is a nonprofit organization dedicated to building a globally accessible public commons of knowledge and culture. CC Provide Creative Commons licenses and public domain tools that give every person and organization in the world a free, simple, and standardized way to grant copyright permissions for creative and academic works; ensure proper attribution; and allow others to copy, distribute, and make use of those works. Creative Commons offers six copyright licenses, based on combinations of the four conditions outlined above.

- [Attribution \(CC BY\)](#)
- [Attribution ShareAlike \(CC BY-SA\)](#)
- [Attribution-NoDerivs \(CC BY-ND\)](#)
- [Attribution-NonCommercial \(CC BY-NC\)](#)
- [Attribution-NonCommercial-ShareAlike \(CC BY-NC-SA\)](#)
- [Attribution-NonCommercial-NoDerivs \(CC BY-NC-ND\)](#)

The following URL links to a guide to use Creativecommons license software:

[https://meta.wikimedia.org/wiki/Open\\_Content\\_-\\_A\\_Practical\\_Guide\\_to\\_Using\\_Creative\\_Commons\\_Licences/The\\_Creative\\_Commons\\_licencing\\_scheme](https://meta.wikimedia.org/wiki/Open_Content_-_A_Practical_Guide_to_Using_Creative_Commons_Licences/The_Creative_Commons_licencing_scheme)

Creativecommons license generator: <https://creativecommons.org/choose/>

### Terms of use:

<https://docs.google.com/document/d/1sAfNj2jQuvSmJg6MSQo5n4qe06scUGqgIBKnB6i6P48/edit?usp=sharing>



**Cite us:**

*By downloading or making this data available to others you agree to the terms of this license and agree to let these terms known to other parties to whom you make this data available and cite us:*

*“Cite “*

# 16. TEMPLATE FROM THE GUADALQUIVIR ESTUARY

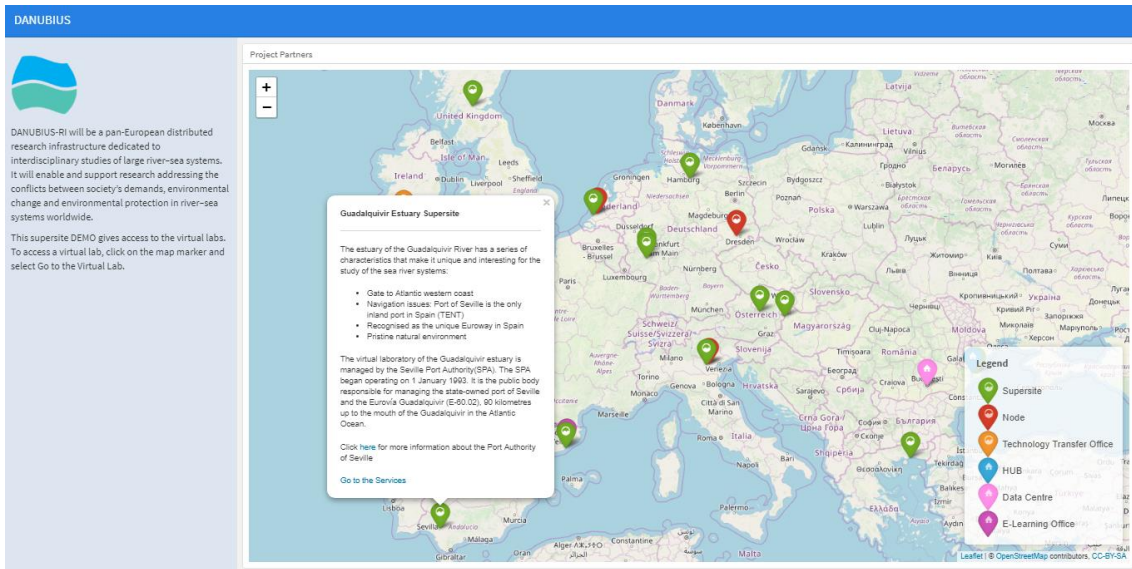
## 1. BRIEF DESCRIPRION OF SUPERSITE

The estuary of the Guadalquivir River has a series of characteristics that make it unique and interesting for the study of the sea river systems:

- Gate to Atlantic western coast.
- Navigation issues: Port of Seville is the only inland port in Spain (TENT).
- Recognized as the unique Euroway in Spain.
- Pristine natural environment.

The virtual laboratory of the Guadalquivir estuary is managed by the Seville Port Authority (SPA). The SPA began operating on 1 January 1993. It is the public body responsible for managing the state-owned port of Seville and the Guadalquivir Euroway(E-60.02), 90 kilometers up to the mouth of the Guadalquivir in the Atlantic Ocean.

Click [here](#) for more information about the Port Authority of Seville.



## 2. DESCRIPTION PAGE

### *Introduction:*



The estuary of the Guadalquivir River has a series of characteristics that make it unique and interesting for the study of the sea river systems:

- Gate to Atlantic western coast.
- Navigation issues: Port of Seville is the only inland port in Spain (TENT).
- Recognized as the unique Euroway in Spain.
- Pristine natural environment.

### About:

*The virtual laboratory of the Guadalquivir estuary is managed by the Port Authority of Seville (PAS). The PAS began operating on 1 January 1993. It is the public body responsible for managing the state-owned port of Seville and the Guadalquivir Euroway (E-60.02), 90 kilometers up to the mouth of the Guadalquivir in the Atlantic Ocean. The following entities collaborate with the PAS.*



### Services:

- Data repositories: Monitoring datasets (water & sediments); Mineralogical & geo-hydrochemical analyses; Publications.
- Expert support: Use and Interpretation of data; Sensor deployment and Calibration; Geodynamic physical modelling.
- Sample collection and provision of data (repositories): Feld stations, Observatory points, Floating platforms, Buoys, etc.
- Evaluation of the aquatic ecosystem health throughout several approaches: Bio / Molecular markers; Presence of microbial species with pathogenic potential; or Transitional model organisms.
- Training services: Workshops; Conferences.



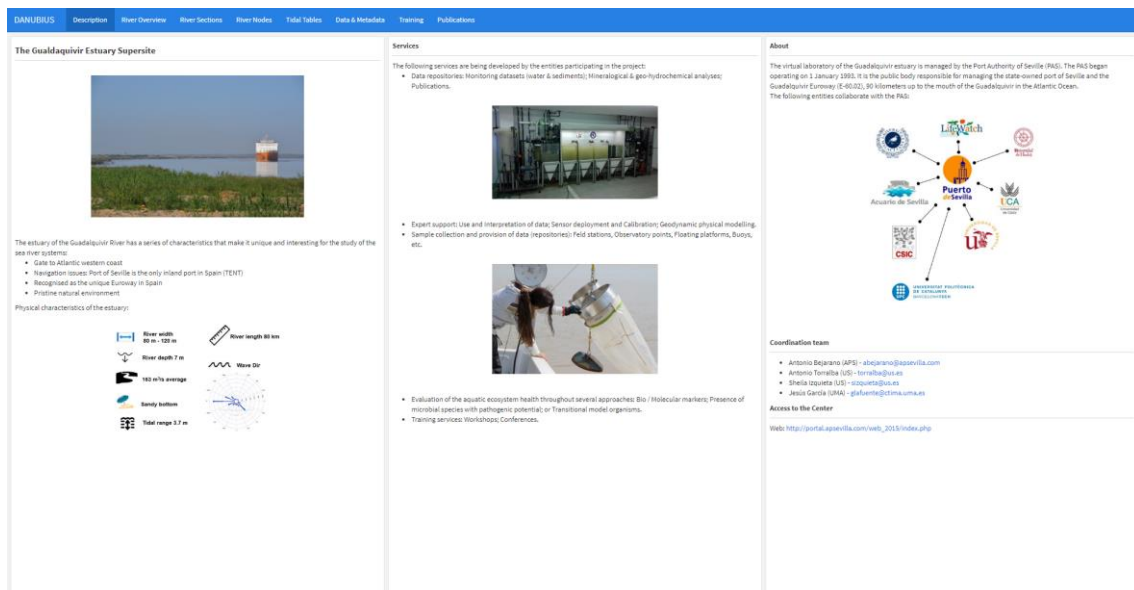
**Coordination team:**

- Antonio Bejarano (APS) - [abejarano@apsevilla.com](mailto:abejarano@apsevilla.com)
- Antonio Torralba (US) - [torralba@us.es](mailto:torralba@us.es)
- Sheila Izquieta (US) - [sizquieta@us.es](mailto:sizquieta@us.es)
- Jesús García (UMA) - [glafuente@ctima.uma.es](mailto:glafuente@ctima.uma.es)
- Javier Delgado (UMA) - [jdcabello@uma.es](mailto:jdcabello@uma.es)

**Access to the Center:**

Web: [http://portal.apsevilla.com/web\\_2015/index.php](http://portal.apsevilla.com/web_2015/index.php)





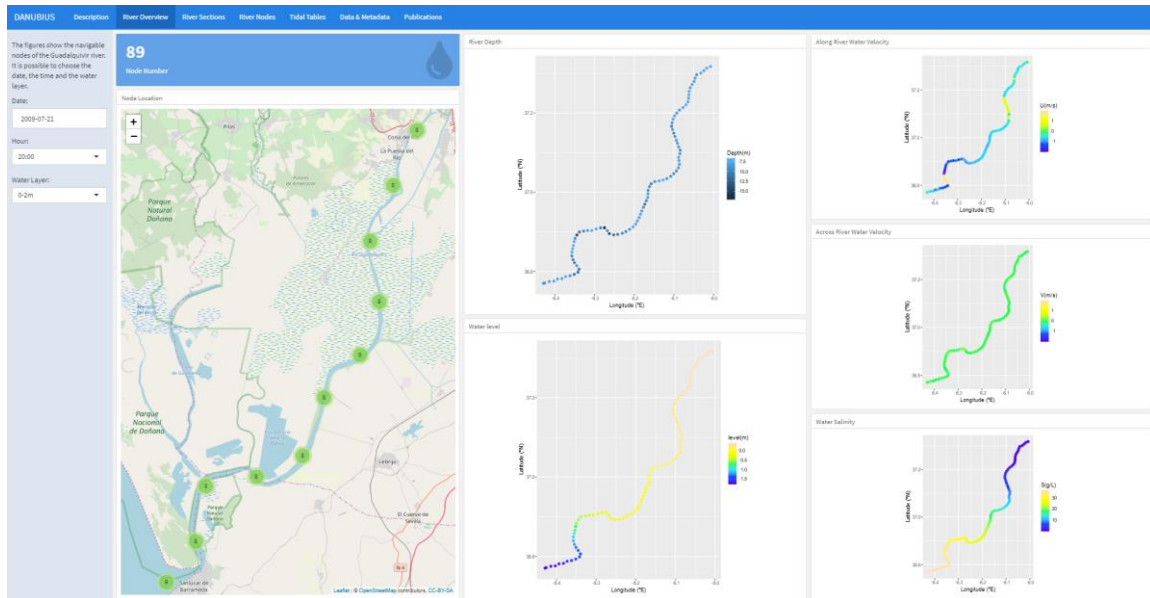
### 3. INTERACTIVE PAGES

#### PAGE 1 – RIVER OVERVIEW

The figures show the navigable nodes of the Guadalquivir river. It is possible to choose the date, the time and the water layer.

#### LAYOUT:

- Column 0 - Left sidebar:
  - Text:
    - The figures show the navigable nodes of the Guadalquivir river. It is possible to choose the date, the time and the water layer.
  - Filter:
    - Date
    - Hour
    - Water Layer
- Column 1
  - Node counter
  - Map
  - Full river with nodes. Popups with longitude, latitude and depth values.
- Column 2
  - Figure: River depth visualization.
  - Figure: River water level visualization.
- Column 3
  - Figure: Along River Water Velocity.
  - Figure: Across River Water Velocity.
  - Figure: Water Salinity.



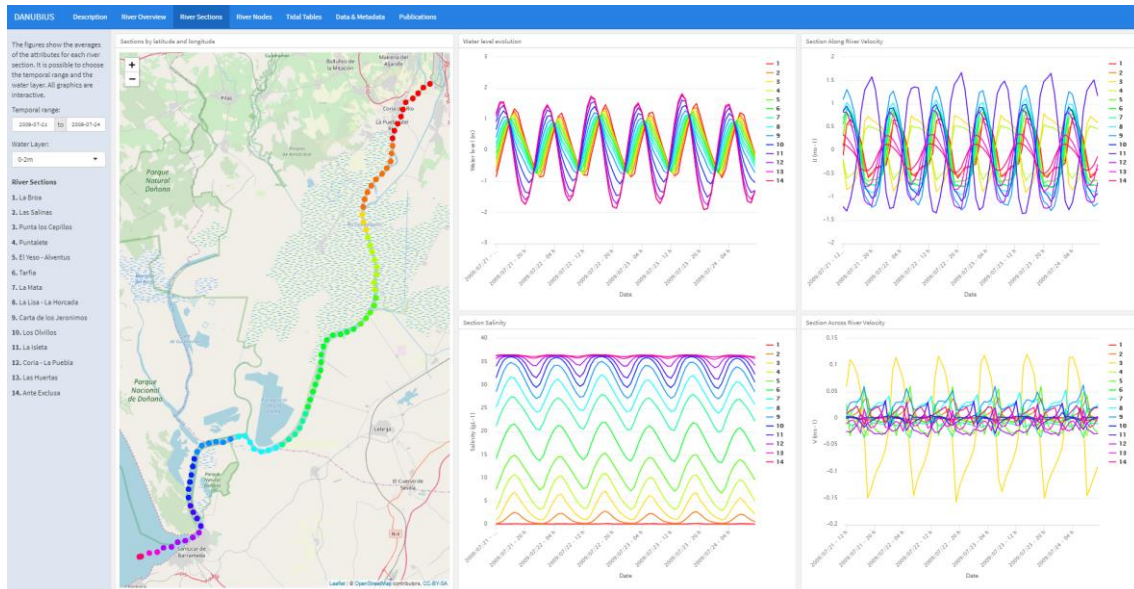
## PAGE 2 – RIVER SECTIONS

The figures show the averages of the attributes for each river section. It is possible to choose the temporal range and the water layer. All graphics are interactive.

### LAYOUT:

- Column 0 - Left sidebar:
  - Text:
    - The figures show the averages of the attributes for each river section. It is possible to choose the temporal range and the water layer. All graphics are interactive.
  - Filter:
    - Temporal range
    - Water Layer
  - Text:
    - River Sections
    - La Broa
    - Las Salinas
    - Punta los Cepillos
    - Puntalete
    - El Yeso - Alventus
    - Tarfia
    - La Mata
    - La Lisa - La Horcada
    - Carta de los Jeronimos
    - Los Olvillos
    - La Isleta
    - Coria - La Puebla
    - Las Huertas

- Ante Exclusa
- Column 1
  - Map
    - Full river with nodes. Popups with longitude, latitude, depth values and section. Colored by sections.
- Column 2
  - Figure: River water level visualization. Colored by sections.
  - Figure: Water Salinity. Colored by sections.
- Column 3
  - Figure: Along River Water Velocity. Colored by sections.
  - Figure: Across River Water Velocity. Colored by sections.



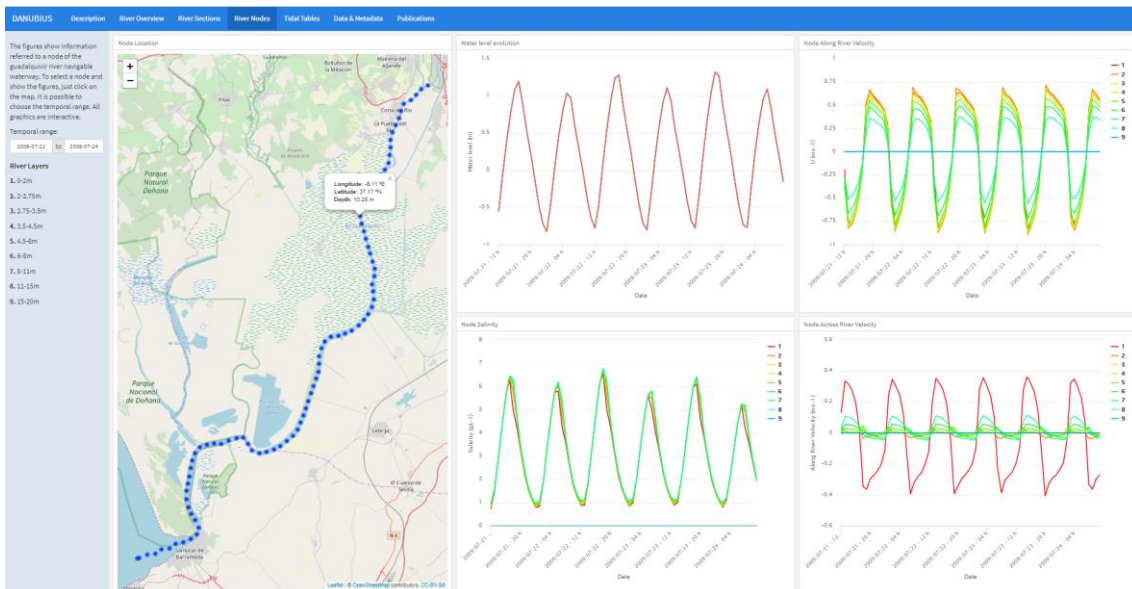
### PAGE 3 – RIVER NODES

The figures show information referred to a node of the Guadalquivir river navigable waterway. To select a node and show the figures, just click on the map. It is possible to choose the temporal range. All graphics are interactive.

#### LAYOUT:

- Column 0 - Left sidebar:
  - Text:
    - The figures show information referred to a node of the Guadalquivir river navigable waterway. To select a node and show the figures, just click on the map. It is possible to choose the temporal range. All graphics are interactive.

- Filter:
  - Temporal range
- Text:
  - River Layers:
    - 0-2m
    - 2-2.75m
    - 2.75-3.5m
    - 3.5-4.5m
    - 4.5-6m
    - 6-8m
    - 8-11m
    - 11-15m
    - 15-20m
- Column 1
  - Map
    - Full river with nodes. Popups with longitude, latitude and depth values.
- Column 2
  - Figure: River water level visualization. Colored by Water Layers.
  - Figure: Water Salinity. Colored by Water Layers.
- Column 3
  - Figure: Along River Water Velocity. Colored by Water Layers.
  - Figure: Across River Water Velocity. Colored by Water Layers.

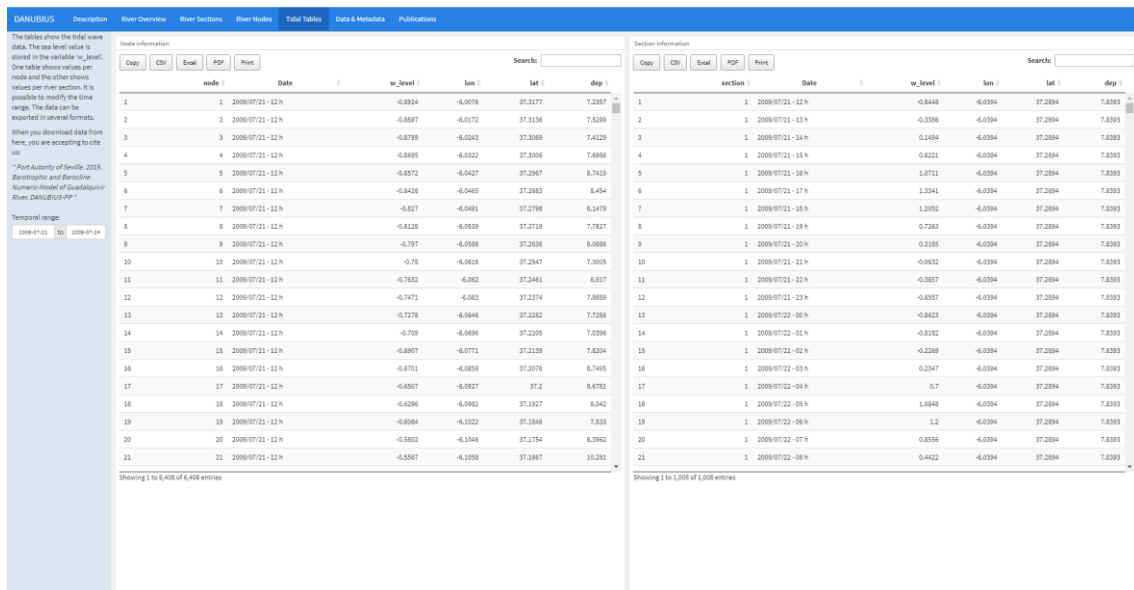


## PAGE 4 – TIDAL TABLES

The tables show the tidal wave data. The sea level value is stored in the variable 'w\_level'. One table shows values per node and the other shows values per river section. It is possible to modify the time range. The data can be exported in several formats.

### LAYOUT:

- Column 0 - Left sidebar:
  - Text:
    - The tables show the tidal wave data. The sea level value is stored in the variable 'w\_level'. One table shows values per node and the other shows values per river section. It is possible to modify the time range. The data can be exported in several formats.
  - Filter:
    - Temporal range
- Column 1
  - Table: Nodes Information
- Column 2
  - Table: Section information



The screenshot displays the DANUBIUS web interface. The left sidebar contains a description of the tidal wave data and a temporal range filter set to 2008-07-21 to 2008-07-24. The main content area is split into two columns. The left column, titled 'Nodes Information', shows a table with columns: node, Date, w\_level, lon, lat, and dep. The right column, titled 'Section Information', shows a table with columns: section, Date, w\_level, lon, lat, and dep. Both tables contain 21 rows of data.

node	Date	w_level	lon	lat	dep
1	2008/07/21-12 h	-0.8924	-6.0076	37.3177	7.2357
2	2008/07/21-12 h	-0.8897	-6.0172	37.3136	7.5209
3	2008/07/21-12 h	-0.8799	-6.0243	37.3069	7.4129
4	2008/07/21-12 h	-0.8695	-6.0322	37.3006	7.6986
5	2008/07/21-12 h	-0.8572	-6.0427	37.2967	8.7429
6	2008/07/21-12 h	-0.8426	-6.0485	37.2883	8.454
7	2008/07/21-12 h	-0.827	-6.0481	37.2798	8.1479
8	2008/07/21-12 h	-0.8128	-6.0539	37.2719	7.7927
9	2008/07/21-12 h	-0.797	-6.0588	37.2638	8.0886
10	2008/07/21-12 h	-0.78	-6.0616	37.2547	7.3005
11	2008/07/21-12 h	-0.7633	-6.062	37.2461	6.917
12	2008/07/21-12 h	-0.7471	-6.063	37.2374	7.6639
13	2008/07/21-12 h	-0.7278	-6.0646	37.2282	7.7288
14	2008/07/21-12 h	-0.709	-6.0666	37.2205	7.0396
15	2008/07/21-12 h	-0.6907	-6.071	37.2139	7.8204
16	2008/07/21-12 h	-0.6761	-6.0689	37.2076	8.7495
17	2008/07/21-12 h	-0.6507	-6.0927	37.2	8.6781
18	2008/07/21-12 h	-0.6296	-6.0922	37.1927	8.042
19	2008/07/21-12 h	-0.6064	-6.1022	37.1846	7.833
20	2008/07/21-12 h	-0.5903	-6.1046	37.1754	8.2922
21	2008/07/21-12 h	-0.5567	-6.1058	37.1667	10.281

section	Date	w_level	lon	lat	dep
1	2008/07/21-12 h	-0.8448	-6.0394	37.2894	7.8393
2	2008/07/21-12 h	-0.8306	-6.0394	37.2894	7.8393
3	2008/07/21-12 h	0.1484	-6.0394	37.2894	7.8393
4	2008/07/21-12 h	0.6221	-6.0394	37.2894	7.8393
5	2008/07/21-12 h	1.0711	-6.0394	37.2894	7.8393
6	2008/07/21-12 h	1.0341	-6.0394	37.2894	7.8393
7	2008/07/21-12 h	1.2052	-6.0394	37.2894	7.8393
8	2008/07/21-12 h	0.7263	-6.0394	37.2894	7.8393
9	2008/07/21-12 h	0.3185	-6.0394	37.2894	7.8393
10	2008/07/21-12 h	-0.3622	-6.0394	37.2894	7.8393
11	2008/07/21-12 h	-0.8857	-6.0394	37.2894	7.8393
12	2008/07/21-12 h	-0.8557	-6.0394	37.2894	7.8393
13	2008/07/21-12 h	-0.8623	-6.0394	37.2894	7.8393
14	2008/07/21-12 h	-0.8182	-6.0394	37.2894	7.8393
15	2008/07/21-12 h	-0.2269	-6.0394	37.2894	7.8393
16	2008/07/21-12 h	0.2347	-6.0394	37.2894	7.8393
17	2008/07/21-12 h	0.7	-6.0394	37.2894	7.8393
18	2008/07/21-12 h	1.0948	-6.0394	37.2894	7.8393
19	2008/07/21-12 h	1.2	-6.0394	37.2894	7.8393
20	2008/07/21-12 h	0.8556	-6.0394	37.2894	7.8393
21	2008/07/21-12 h	0.4422	-6.0394	37.2894	7.8393



## 4. DATA and METADATA

### *Data Format and Structure*

#### Data description:

The simulated models contain two years (2008 - 2009) of data with a sampling rate of 1 hour. There is a barocline and a bathymetric model. These models are combined in a finite elements mesh in NetCDF format of the Guadalquivir river estuary.

The mesh is composed by 7010 nodes, 89 referred to the navigable channel of the river. The navigable channel of the Guadalquivir is 78 km long; between one node and the next there is an average distance of 1km. Each node has 3 geographical properties (“latitude”, “longitude” and “deep”).

There are 3 variables (“Along River Velocity”, “Across River Velocity” and “Salinity”) that depend on each node and the “Water Layer”. “Water Layer” refers to the various layers that the depth of the river can be divided.

More information about barocline and barotropic models:  
[http://snowball.millersville.edu/~adecaria/ESCI342/esci342\\_lesson11\\_circulation.pdf](http://snowball.millersville.edu/~adecaria/ESCI342/esci342_lesson11_circulation.pdf)

#### Data format:

NetCDF

#### Data structure:

- 7010 Nodes mesh (89 navigation nodes):
  - Nodes: latitude, longitude and depth
    - Along River Velocity ~ [node, water layer, time]
    - Across River Velocity ~ [node, water layer, time]
    - Salinity ~ [node, water layer, time]
    - Water level ~ [node, time]

### *Metadata*

- Dimensions
- Variables
  - Each listed item describes the following information
    - Data type
    - Units
    - Content description
    - Missing data values

(The Dimension and Variables metadata is available and full filled but it is not shown in the DEMO)

- Global Attributes
  - Conventions: CF-1.4
  - Authors
    - Author/s name:
    - License: (Link to license)
    - Terms of Use: (Link to terms of use)



Preparatory Phase for the pan-European  
Research Infrastructure DANUBIUS-RI  
“The International Centre for advanced  
studies on river-sea systems”

- Cite us: Port Authority of Seville. 2019. Barotropic and Barocline Numeric-Model of Guadalquivir River. DANUBIUS-PP
- References: Model info: <http://oceano.uma.es/>
- Source: MOGUASHY LR Barotropic model v2.0
- History: created on 2019-06-20 13:57:48 MET
- Institute: GOFIMA. Grupo de Oceanografía Física de la Universidad de Málaga
- Project: DANUBIUS
- Contact: email: [jdcabello@uma.es](mailto:jdcabello@uma.es)
- Comments: Data restriction: for academic research use only.
- Software version
- Model
  - Sampling Rate: 1 hour
  - Date: 01/01/2008 00:00h – 02/09/2009 14:00h
  - Title: Salidas SIPORT21, Low Resolution (LR) grid, fondo APS2015\_Agosto2017\_+0.3+0.18/80xdist\_Bon, marea astronómica + descarga constante desde Alcala 50 m3/s
  - Global Unique identifier (e.g.: DOI, PID, ...):
  - Metadata link
- Keywords

#### **Term of use - License - Cite Us**

The Data remains the property of the Port Authority of Seville, the University of Seville, and the University of Malaga.

This data distributed "AS IS" under this License in the hope that it will be useful, but in order that the University protects its assets for the benefit of its educational and research purposes, the University makes clear that no condition is made or to be implied, nor is any warranty given or to be implied, as to the accuracy of the Data, or that it will be suitable for any particular purpose or for use under any specific conditions. Furthermore, the University disclaims all responsibility for the use which is made of the Data. It further disclaims any liability for the outcomes arising from using the Data.

These data have been produced by the SHYFEM\* model adapted to the Guadalquivir estuary by the Physical Oceanography Group of the University of Malaga (GOFIMA). GOFIMA is thanked for the availability of these data

\*SHYFEM has been developed at ISMAR-CNR (Institute of Marine Sciences - National Research Council; Umgiesser, 2004.

*Umgiesser, G., Melaku Canu, D., Cucco, A., Solidoro, C., 2004. A finite element model for the Venice Lagoon. Development, set up, calibration and validation. Journal of Marine System 51, 123–145.*



# Preparatory Phase for the pan-European Research Infrastructure DANUBIUS-RI “The International Centre for advanced studies on river-sea systems”

**DANUBIUS** Description River Overview River Sections River Nodes Total Tables Data & Metadata Training Publications

### Data Format and Structure

**Data description:**  
The simulated models contain two years (2008 - 2009) of data with a sampling rate of 1 hour. There is a baroclinic and a bathymetric model. These models are combined in a finite elements mesh in NetCDF format of the Guadalquivir river estuary.  
The mesh is composed by 7102 nodes, 89 referred to the navigable channel of the river. The navigable channel of the Guadalquivir is 78 km long, between one node and the next there is an average distance of 1km. Each node has 3 geographical properties ("latitude", "longitude" and "depth").  
There are 3 variables ("Along River Velocity", "Across River Velocity" and "Salinity") that depend on each node and the "Water Layer"/"Water Layer" refers to the various layers that the depth of the river can be divided.  
More information about baroclinic and bathymetric models [http://snowball.millersville.edu/~adecaria/ESCI342/lec342\\_jeason11\\_circulation.pdf](http://snowball.millersville.edu/~adecaria/ESCI342/lec342_jeason11_circulation.pdf)

**Data format:**  
netCDF

**Data structure:**

- 7102 Nodes mesh (89 navigation nodes)
  - Nodes: latitude, longitude and depth
  - Along River Velocity - [node, water layer, time]
  - Across River Velocity - [node, water layer, time]
  - Salinity - [node, water layer, time]
  - Water level - [node, time]

**Metadata:**

- Conventions: CF-1.4
- Authors
  - Cite as:  
These data have been produced by the SHYFEM model adapted to the Guadalquivir estuary by the Physical Oceanography Group of the University of Malaga (GOFIMA). GOFIMA is thanked for the availability of these data. SHYFEM has been developed at ISMAR-CNR (Institute of Marine Sciences - National Research Council); Ungerszer, 2004

Ungerszer, G., Hekko Camu, D., Cucco, A., Solidoro, C., 2004. A finite element model for the Venice Lagoon: Development, set up, calibration and validation. *Journal of Marine System* 51, 223-245.

- Reference Model info: <http://icecero.um.es/>
- Source: HYCOM/SHYFEM Bathymetric model v2.0
- History: created on 2019-09-20 13:57:48 HET
- Institute: GOFIMA, Grupo de Oceanografía Física de la Universidad de Málaga
- Project: DANUBIUS
- Contact: email: [jstabelle@um.es](mailto:jstabelle@um.es)
- Comments: Data restriction for academic research use only.
- Software version:

**Model:**

- Sampling Rate: 1 hour
- Date: 15/01/2008 00:00h - 02/08/2009 14:00h
- Title: Salidas SHYFEM2, Low Resolution (LR) grid, fondo APS2013\_Agosto2017\_v=0.3=0.18/30diod\_Bon, marea astronomica + descarga constante desde Alcalá 50 m3/s

**Term of use - License - Cite file:**

The Data remains the property of the Port Authority of Seville, the University of Seville, and the University of Malaga. This data distributed "AS IS" under this license in the hope that it will be useful, but in order that the University protects its assets for the benefit of its educational and research purposes, the University makes clear that no condition is made or to be implied, nor is any warranty given or to be implied, as to the accuracy of the Data, or that it will be suitable for any particular purpose or for use under any specific conditions. Furthermore, the University disclaims all responsibility for the use which is made of the Data. It further disclaims any liability for the outcomes arising from using the Data.

These data have been produced by the SHYFEM model adapted to the Guadalquivir estuary by the Physical Oceanography Group of the University of Malaga (GOFIMA). GOFIMA is thanked for the availability of these data.

SHYFEM has been developed at ISMAR-CNR (Institute of Marine Sciences - National Research Council); Ungerszer, 2004

Ungerszer, G., Hekko Camu, D., Cucco, A., Solidoro, C., 2004. A finite element model for the Venice Lagoon: Development, set up, calibration and validation. *Journal of Marine System* 51, 223-245.

## 5. eTRAINING

List of training courses dedicated to the services of the center, if any

## 6. PUBLICATIONS

[1] - Torralba Silgado, A. J., Gutierrez-Rumbao, J., Peral, J. M., Daza Rebollo, D., Rodríguez-Serrano, A., Hidalgo, E., ... & Escudero Santana, A. (2015). Smart ICTs for the enhancement of traffic logistics in the Port of Seville.

[2] - Delgado, J., Moreno-Navas, J., Pulido, A., García-Lafuente, J., Calero Quesada, M. C., & García, R. (2017, April). Modelling extreme climatic events in Guadalquivir Estuary (Spain). In EGU General Assembly Conference Abstracts (Vol. 19, p. 19551).

[3] - Ruiz-Aguilar, J. J., Turias, I. J., Cerbán, M., Jiménez-Come, M. J., González, M. J., & Pulido, A. (2016). Time analysis of the containerized cargo flow in the logistic chain using simulation tools: the case of the Port of Seville (Spain). *Transportation research procedia*, 18, 19-26.

[4]- 3D hydrodynamic model as a tool for more efficient port management and operations. ECSA 56 – Coastal systems in transition from a ‘natural’ to an ‘anthropogenically modified’ 4-7 September 2016. Bremen, Germany

[5] - Modelling sea level rise and extreme climatic events in Guadalquivir Estuary (Spain). ECSA 57 - Changing estuaries, coasts and shelf systems - Diverse threats and opportunities. 3-6 September 2018. Perth, Australia.

[6] - Modeling changes in the tidal propagation and its implication for vessel navigation in Guadalquivir Estuary (Spain). ECSA 57 - Changing estuaries, coasts and shelf systems - Diverse threats and opportunities. 3-6 September 2018. Perth, Australia.





Preparatory Phase for the pan-European  
Research Infrastructure DANUBIUS-RI  
“The International Centre for advanced  
studies on river-sea systems”

[7] - Hydrodynamic changes in Guadalquivir Estuary (Spain) due to a hypothetical tidal marsh restoration project. ECSA 57 - Changing estuaries, coasts and shelf systems - Diverse threats and opportunities. 3-6 September 2018. Perth, Australia.

[8] - Modeling tidal propagation in Guadalquivir Estuary (Spain), 2019. Model output application to vessel navigation. Smart Rivers 2019 Conference September 30 - October 3, 2019 Centre de Congrès Lyon FRANCE

The screenshot shows the 'Publications' tab of the DANUBIUS-RI website. The page title is 'List of Publications'. The list contains eight entries, each with a reference number in brackets and a brief description of the publication. The entries are:

- [2] - Torralba Silgado, A. J., Gutierrez-Rumbao, J., Peral, J. M., Doza Rebollo, D., Rodriguez-Serrano, A., Hidalgo, E., ... A. Escuders Santana, A. (2015). Smart ICTs for the enhancement of traffic logistics in the Port of Seville.
- [3] - Delgado, J., Moreno-Heras, J., Pulido, A., Garcia-Lafrer, J., Calero-Quesada, M. C., & Garcia, B. (2017, April). Modelling extreme climatic events in Guadalquivir Estuary (Spain). In ESU General Assembly Conference Abstracts (Vol. 18, p. 19551).
- [4] - Ruiz-Aguller, J. J., Turiso, I. J., Carballa, M., Jimenez-Come, M. J., Gonzalez, M. A., & Pulido, A. (2016). Time analysis of the containerized cargo flow in the logistic chain using simulation tools: the case of the Port of Seville (Spain). Transportation research proceedings, 18, 19-26. <https://doi.org/10.1016/j.trpro.2016.12.003>
- [5] - 3D hydrodynamic model as a tool for more efficient port management and operations. ECSA 56 - Coastal systems in transition from a 'natural' to an 'anthropogenically modified' 4-7 September 2016, Bremen, Germany
- [6] - Modelling sea level rise and extreme climatic events in Guadalquivir Estuary (Spain). ECSA 57 - Changing estuaries, coasts and shelf systems - Diverse threats and opportunities. 3-6 September 2018, Perth, Australia.
- [6] - Modelling changes in the tidal propagation and its implication for vessel navigation in Guadalquivir Estuary (Spain). ECSA 57 - Changing estuaries, coasts and shelf systems - Diverse threats and opportunities. 3-6 September 2018, Perth, Australia.
- [7] - Hydrodynamic changes in Guadalquivir Estuary (Spain) due to a hypothetical tidal marsh restoration project. ECSA 57 - Changing estuaries, coasts and shelf systems - Diverse threats and opportunities. 3-6 September 2018, Perth, Australia.
- [8] - Modeling tidal propagation in Guadalquivir Estuary (Spain), 2019. Model output application to vessel navigation. Smart Rivers 2019 Conference September 30 - October 3, 2019 Centre de Congrès Lyon FRANCE



Preparatory Phase for the pan-European  
Research Infrastructure DANUBIUS-RI  
“The International Centre for advanced  
studies on river-sea systems”



**European  
Commission**

This project has received funding from the European Union's  
Horizon 2020 Research and Innovation Programme under  
Grant Agreement No 739562