Towards ENVRI Community International Winter School
DATA FAIRness
Webinar Programme – July / September 2020

ENVRI FAIR

Workflows Orchestration and Execution

Speakers: Nicola Fiore (nicola.fiore@lifewatch.eu), LifeWatch ERIC Service Centre
Lucia Vaira (lucia.vaira@lifewatch.eu), LifeWatch ERIC Service Centre

LifeWatch ERIC

ENVRI-FAIR has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 824068

Outline

Theoretical part

- Workflows and examples
  - Human-Centric vs. System-Centric Workflows
  - Automated vs. Manual Workflows
- Scientific Workflows
  - Historical background
  - The workflow lifecycle
- Orchestration and Choreography
- Scientific workflow systems
  - User Requirements
  - Technical Requirements
  - Why a GUI

Practical part

- Phytoplankton VRE: the showcase (objectives, steps, results) in Taverna
- Introduction on Node-RED (main features and functionalities)
- Phytoplankton VRE on Node-RED
- Other examples in Node-RED

ENVRI FAIR

Who is the audience?

1. Which is your background?

Staff of RI	9
Academic	8
Technical staff	12
PhD Student	1

https://www.menti.com/

54 92 70

On a laptop: visit https://www.menti.com/ and put the code **549270**
On a smartphone: scan the QR code and answer

Business Workflows

- Business workflow management and business process modelling are mature research areas, whose roots go far back to the early days of office automation systems.

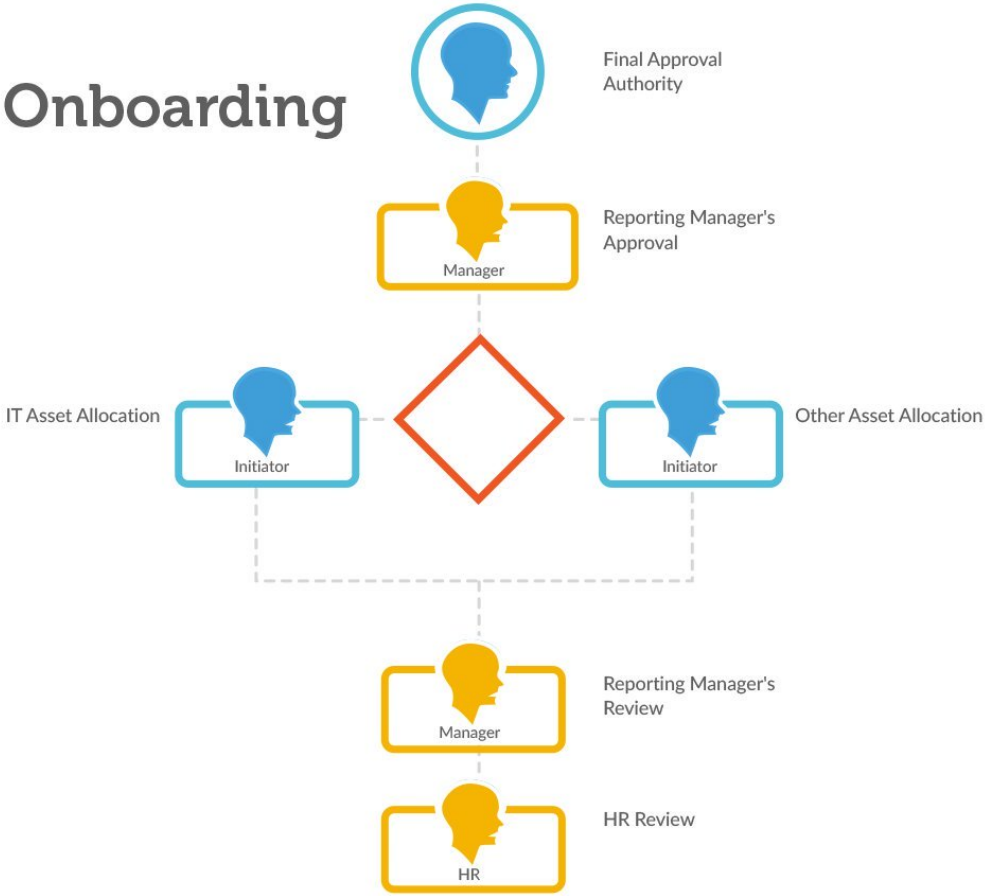
- A Workflow is a sequence of tasks that processes a set of data. Workflows occur across every kind of business and industry. Anytime data is passed between humans and/or systems, a workflow is created. Workflows are the paths that describe how something goes from being undone to done, or raw to processed.

Workflows hide in many places:

- If you have a lot of emails you mindlessly pass down an invisible chain, that's a workflow.
- If you print the same form over and over again, that's a workflow.
- If you find yourself turning to a spreadsheet to organize dynamic data, that's a workflow.
- If you find your work is getting held up because someone else isn't doing their job well, that's a workflow.

Business Workflows

Employee Onboarding Workflow



What is not a Workflow?

- If data isn't moving, you don't have a workflow.
- For example, if you are managing a list of unconnected tasks (walk the dog, go to the grocery store, pick up the dry cleaning), this isn't a workflow, but task management.
- For it to be a workflow, the tasks have to be connected in a way to be a part of something bigger.

What is not a Workflow?

- **Workflow is not a Process**
- Workflows only describe the sequence of tasks.
- A process is a broader term that also encompasses the data, forms, reports, and notifications required to get an item from start to finish in a structured environment.
- For example, the workflow for purchase orders might be Initiator => Manager Approval => Procurement Processing.
- But the process also involves a data set of approved vendors to choose from, the individual sequential number assigned to the purchase order, how procurement is notified, the budget available, and many more factors.

What is not a Workflow?

- **Workflow is not a Checklists**
- A checklist is an elementary version of a workflow.
- Checklists only work for processes and projects but often lack the ability to share across team members.
- Checklists also make it difficult to track items that need to go back to an earlier stage in a workflow.
- Checklists do a poor job processing workflows that are conditional on certain data.
- For example, if you are making marketing campaigns, but you want to follow a different workflow based on what platform will be used to distribute the campaign, you would need to have as many checklists as you have platforms. Whereas with more sophisticated workflow, you can handle all the items in a single workflow.

ENVRI FAIR

Human-Centric vs. System-Centric Workflows

- In **human-centric workflows**, most of the tasks are assigned to humans. These might require approving data, creating something new, or double-checking information.
- In **system-centric workflows**, most of the tasks are done by a machine and require little to no human involvement.
- For example, to create a financial report, a workflow might be triggered at the same time every month to grab certain data from different systems, parse it into a report, and email the report to all the stakeholders. A system can perform all of these tasks.
- There are also document-centric workflows where the entire workflow is built around a document. A good example is a contract for leasing some office space. Everything that happens as a part of the workflow needs to be added or modified on the document and the end result should be a contract that correctly captures all the data in the workflow including digital signatures.

Automated vs. Manual Workflows

- In a manual workflow, a human is responsible for pushing each item from one task to another.
- For example, when an employee fills out a reimbursement claim, she must email it to her manager for approval. After approval, she must email it to the finance department.
- The finance department must go into the software and schedule a payment and then email the employee to say it is complete.
- In an automated workflow, when a human completes a task, she is not responsible for passing the data on to the next task. The workflow is programmed to handle this. The system manages the flow of tasks including notifications, deadlines, and reminders.
- In the same reimbursement example, the employee might fill out a form and hit a submit button. It would automatically trigger a notification for the manager to review it and click Approve.
- This would automatically take it to the finance team for processing, or if the amount is small enough, it would trigger a task to release the payments and send an automated email to the employee.

Benefits of the Automated Workflows

- Tracking items are much easier
- Eliminating redundant tasks
- Improving efficiency
- Simplifying delegation of tasks
- Reducing processing time
- Giving greater visibility
- Establishing accountability

Scientific Workflow

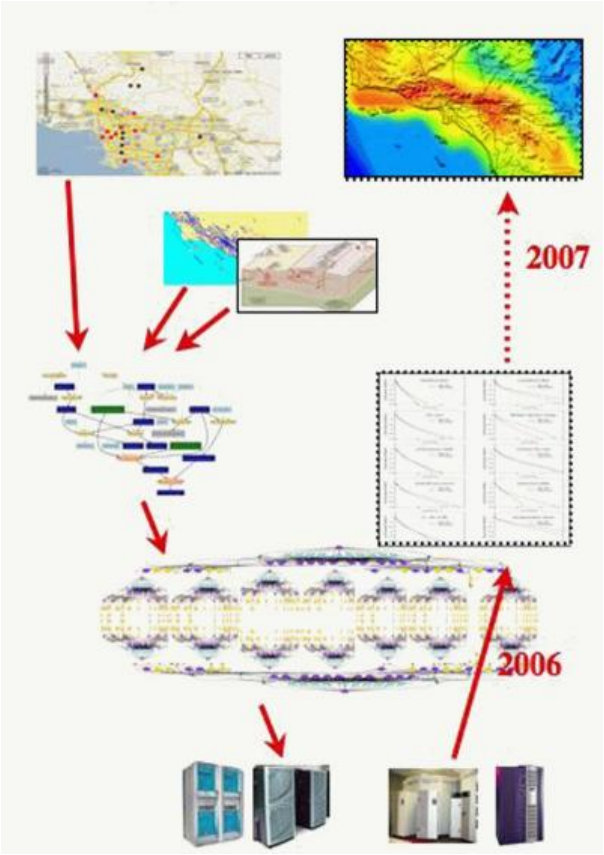
- ***A scientific workflow is the description of a process for accomplishing a scientific objective, usually expressed in terms of tasks and their dependencies. (B. Ludascher et alt., 2009)***

- Scientific workflows allow users to easily express multi-step computational tasks.

- Goal: automate a scientist's repetitive data management and analysis tasks

Typical Phases:

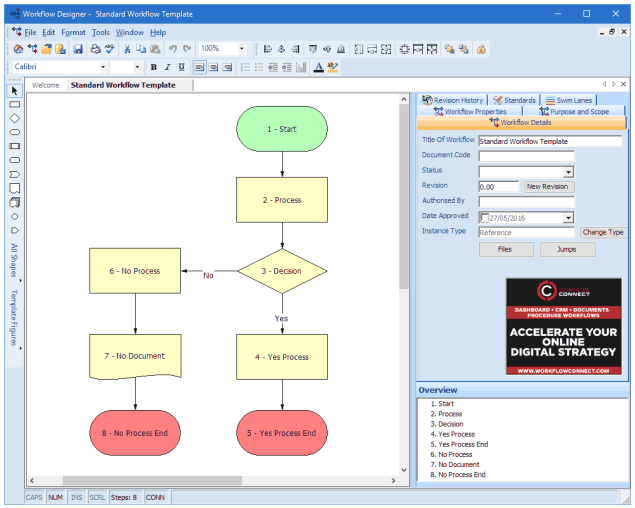
- Acquisition, integration, reduction, visualization, and publication (e.g., in a shared database) of scientific data.
- e.g. Retrieve data from a catalogue or an instrument, reformat the data, and run an analysis.



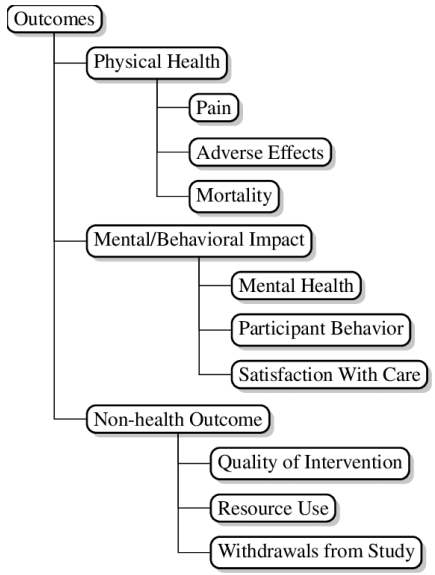
ENVRI FAIR

Scientific Workflow

- The tasks of a scientific workflow are organized (at **design time**) and orchestrated (at **runtime**) according to dataflow and possibly other dependencies as specified by the workflow designer.

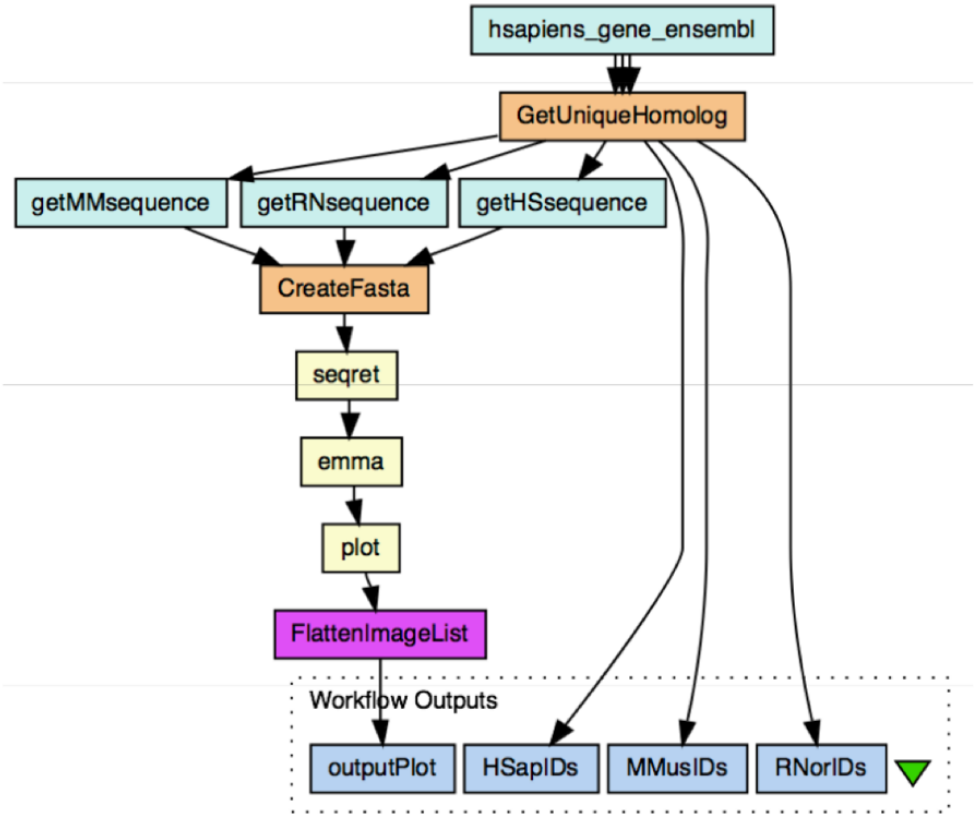


Workflows can be designed visually, e.g., using block diagrams



or textually using a domain-specific language

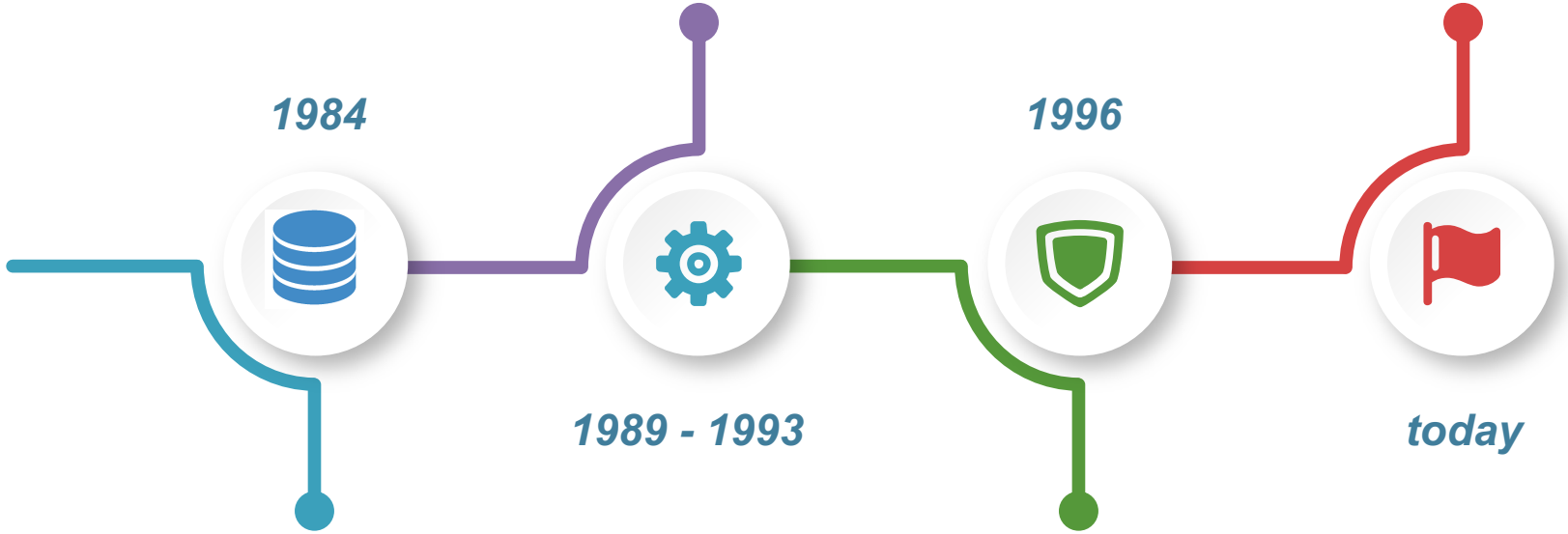
Scientific Workflow Example



- Example workflow represented in the Taverna workflow system.

- This workflow extracts gene IDs from human chromosome 22 with mappings to disease functions and homologues in mouse and rat; fetches base pairs of the associated DNA sequences; combines the sequences into a FASTA file; performs a multiple sequence alignment; and renders the result. The workflow uses three soap lab-based analysis operations (seqret, emma, plot) that run on the EBI compute cluster.

Historical Background



1984

Scientific Database

The database community realize that scientific data management has different characteristics from more traditional business data management

1989 - 1993

Early Works

- MOOSE data model
- FOX query language
- ZOO experiment management environment

1996

Born of the term 'Scientific workflow'

WASA, a Workflow-based Architecture for Scientific Applications

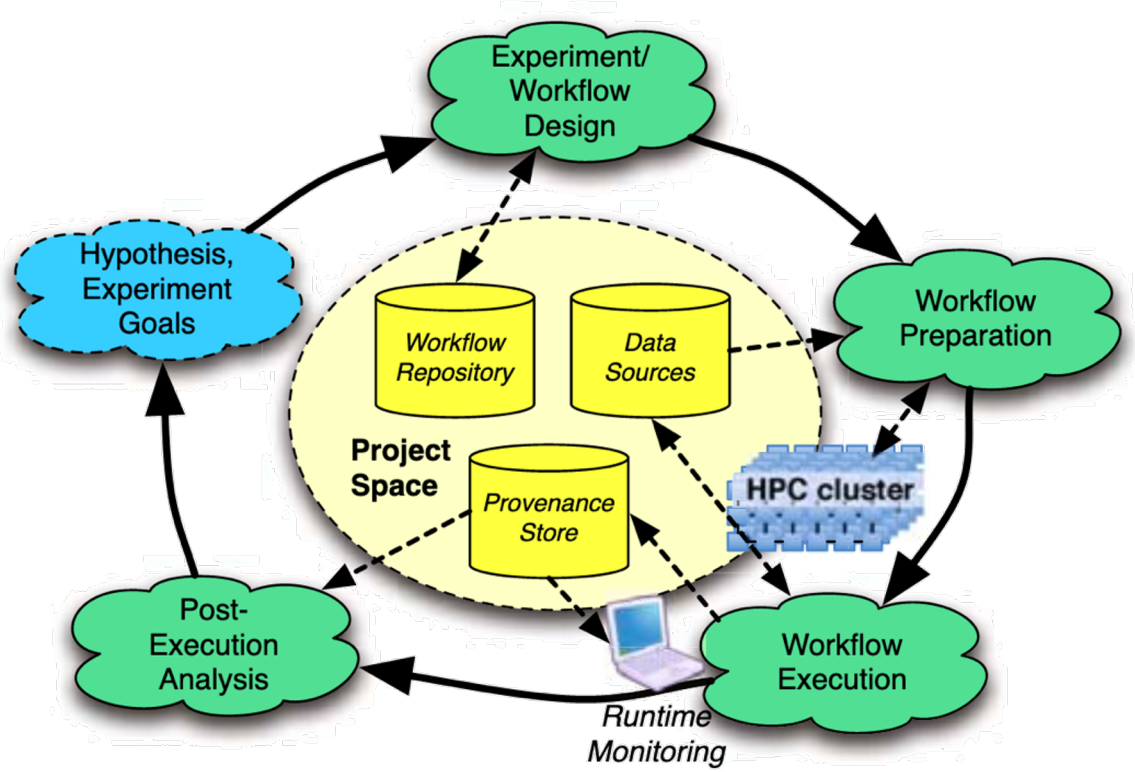
today

e-Science as a paradigm

Computational techniques and tools from the computational sciences, distributed and high-performance computing, databases, data analysis, visualization, sharing, and collaboration

ENVRI FAIR

The Scientific Workflow Life Cycle



(B. Ludascher et al., 2009)

A quick poll

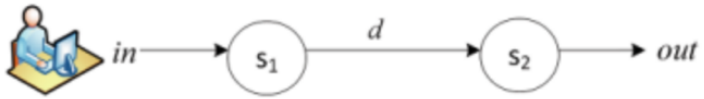
2. Using workflows in biodiversity research is limited but on the rise. If you've worked with scientific workflows, which challenges did you face?

https://www.menti.com/

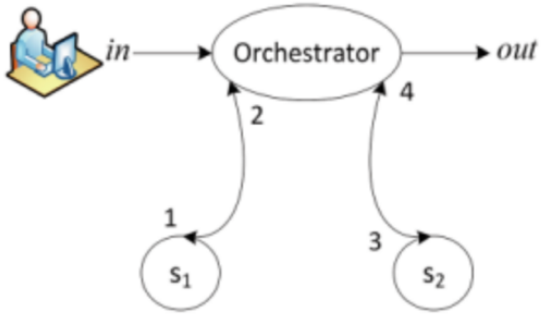
54 92 70

On a laptop: visit https://www.menti.com/ and put the code **549270**
On a smartphone: scan the QR code and answer

Orchestration and Choreography

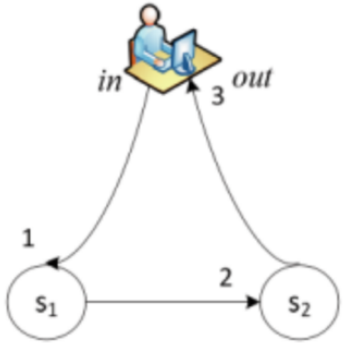


(a) Workflow



1 : Input *in* given to service s_1
2 : Service s_1 produces data *(d) that is* passed to orchestrator
3 : Orchestrator passes data *(d)* to service s_2
4 : Service s_2 sends result *out* to orchestrator

(b) Classical Orchestration Approach



1 : Input *in* given to service s_1 by user
2 : Service s_1 produces data *(d) that is* passed to service s_2
3 : Service s_2 sends result *out* to user

(c) Classical Choreography Approach

Scientific workflow systems - User Requirements

- **Design tools** – especially for non-expert users
  Need to look into how scientists define processes
- **Ease of use** – fairly simple user interface having more complex features hidden in background
- **Reusable generic features**
- **Generic enough** to serve different communities but **specific enough** to serve one domain
- **Extensibility for the expert user** – almost a visual programming interface
- **Registration and publication** of data products and “process products” (workflows); Provenance

Scientific workflow systems - Technical Requirements

- Error detection and recovery from failure
- Logging information for each workflow
- Allow data-intensive and compute-intensive tasks (maybe at the same time)
- Data management/integration
- Allow status checks and on the fly updates
- Visualization
- Semantics and metadata based dataset access
- Certification, trust, security

Scientific workflow systems - Why a GUI

- No need to learn a programming language
- Visual representation of what workflow does
- Allows you to monitor workflow execution
- Enables user interaction
- Facilitates sharing workflows

Scientific workflow systems (SWFSs)

https://taverna.incubator.apache.org/

https://galaxyproject.org/

https://nodered.org/

https://kepler-project.org/

ENVRI FAIR

A quick poll

3. Which orchestrators do you know or you have ever used?

https://www.menti.com/

54 92 70

On a laptop: visit https://www.menti.com/ and put the code **549270**
On a smartphone: scan the QR code and answer

ENVRI FAIR

A quick poll

4. Which programming language(s) do you know?

https://www.menti.com/

54 92 70

On a laptop: visit https://www.menti.com/ and put the code **549270**
On a smartphone: scan the QR code and answer

Take a break!

We will back in 5 mins

Outline

Theoretical part

- Workflows and examples
  - Human-Centric vs. System-Centric Workflows
  - Automated vs. Manual Workflows
- Scientific Workflows
  - Historical background
  - The workflow lifecycle
- Orchestration and Choreography
- Scientific workflow systems
  - User Requirements
  - Technical Requirements
  - Why a GUI

Practical part

- Phytoplankton VRE: the showcase (objectives, steps, results) in Taverna
- Introduction on Node-RED (main features and functionalities)
- Phytoplankton VRE on Node-RED
- Other examples in Node-RED

Phytoplankton Virtual Research Environment

- The **e-Biodiversity Research Institute of LifeWatch Italy** has realised the **Phytoplankton Virtual Research Environment** (Phyto VRE), a collaborative working environment supporting researchers to address basic and applied studies on phytoplankton ecology. The Phyto VRE provides the IT infrastructure enabling researchers to obtain, share and analyse phytoplankton data.

- In order to facilitate the computation of phytoplankton traits and to investigate the distribution patterns, the **Ecology Laboratory of the University of Salento** and **LifeWatch Italy** designed and developed a workflow that allows automating a set of operations written in R language.

UNIVERSITÀ DEL SALENTO

LifeWatch ITALIA

ENVRI FAIR

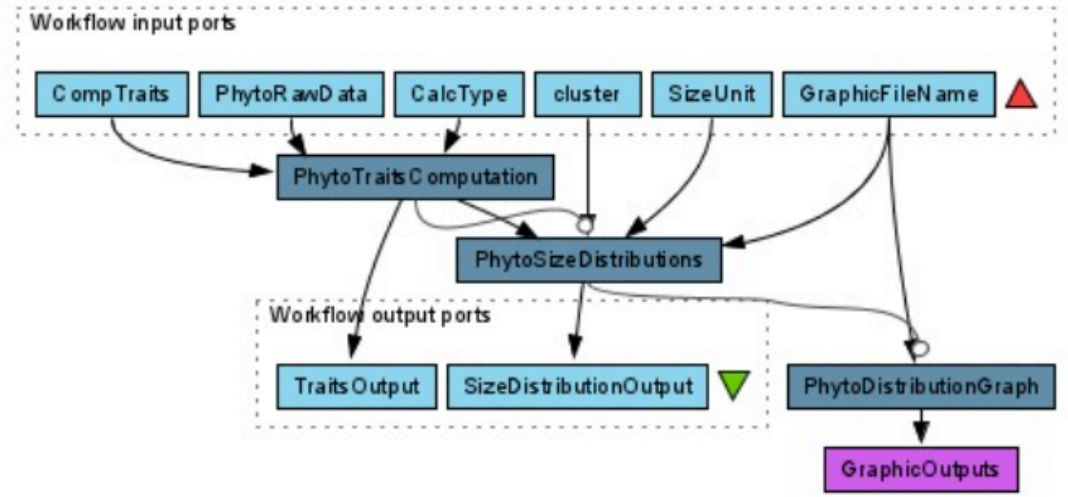
Phytoplankton traits-based analysis SHOWCASE

OBJECTIVE

Objective: facilitate the computation of phytoplankton traits and investigate the distribution patterns.

The workflow is composed by three R scripts:

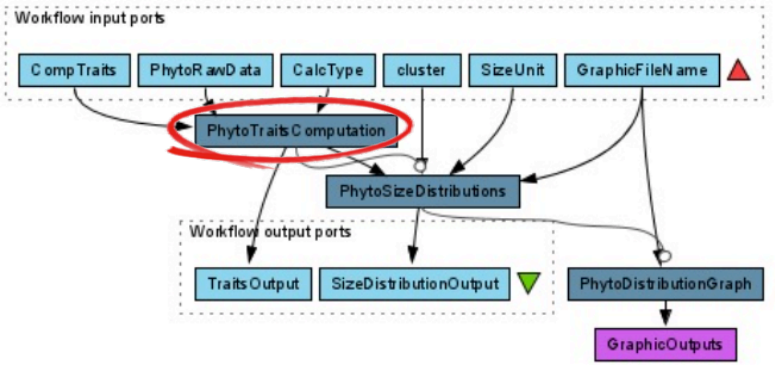
- Phytoplankton Traits Computation
- Phytoplankton Size Distributions
- Phyto Distribution Graph



Phytoplankton traits-based analysis SHOWCASE

STEPS

PhytoTraitsComputation: it computes morphological and demographic traits, such as hidden dimension, biovolume, surface area, surface-volume ratio, cell carbon content, density, carbon content and total biovolume.



Inputs

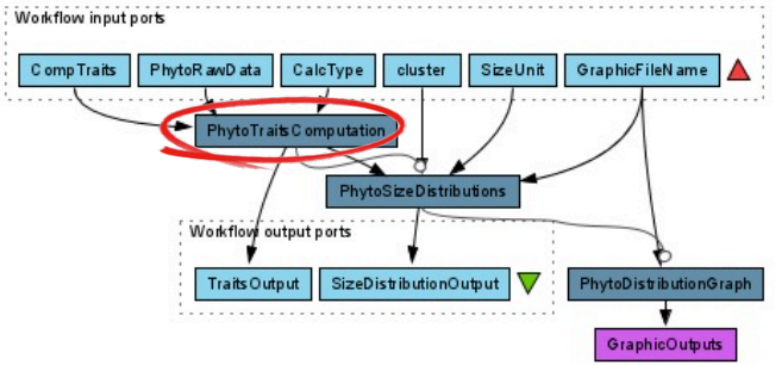
- CompTraits: traits to be computed, i.e., Biovolume (HD, hidden dimension and BV, biovolume), Surface Area (SA), Surface/Volume ratio (SV), Cells/Liter (CL), Biovolume/Liter (BVL), Carbon content (CC), Carbon content/Liter (CCL)
- PhytoRawData: the .csv file with raw data, harmonised according to the LifeWatch Italy Data Schema

ENVRI FAIR

Phytoplankton traits-based analysis SHOWCASE

STEPS

- CalcType: the computation type that can be:
  - Simplified (*true*): it approximates the taxon specific-biovolumes computation based on two linear dimensions only, length and width. Mandatory fields: scientific name, measurement remarks, length and width
  - Advanced (*false*): it allows a more accurate estimate of taxon-specific biovolume, but it requires more information. For each shape, at least 2 measured basic linear dimensions need to be provided. Mandatory fields: scientific name, measurement remarks and linear dimensions.



Output

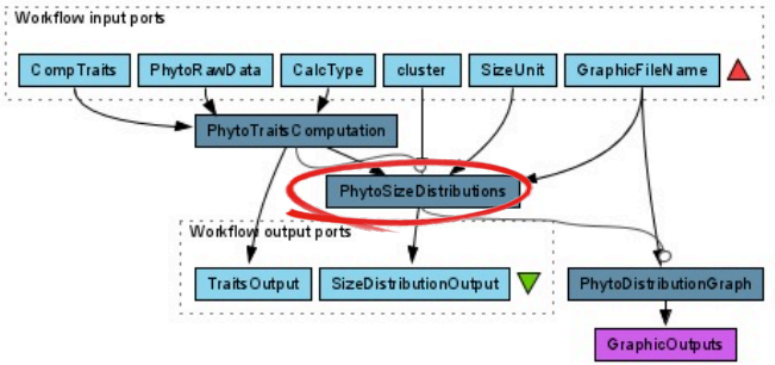
- TraitsOutput: a csv file containing also the computed traits

ENVRI FAIR

Phytoplankton traits-based analysis SHOWCASE

STEPS

PhytoSizeDistributions: it performs Modality (Hartigans' dip test), Normality or LogNormality (Anderson-Darling test, Cramer- von Mises) tests of phytoplankton biovolume (expressed as µm3) or cell carbon content (expressed as pgC*cell-1) distributions, at different levels of data aggregation (i.e. spatial, temporal, taxonomic).



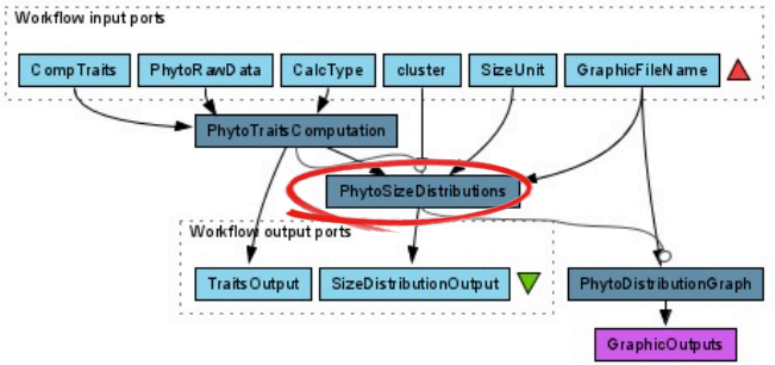
Inputs

- Cluster: the aggregation level for size distributions. It can be:
  - spatial (e.g., eventid, parenteventid, locality, country, Eunis habitat type-name)
  - temporal (e.g., day, month, year)
  - taxonomic (e.g., Phylum, Order, scientific name)

Phytoplankton traits-based analysis SHOWCASE

STEPS

- SizeUnit: the morphological trait that will be used to perform Modality, Normality or LogNormality tests of distributions (BV for distributions based on Biovolume or CellCC for distributions based on Cell carbon content)
- GraphicFileName: the name that will be used to create the pdf distribution file



- TraitsOutputFile: the csv file that has been produced as output in the previous step

Output

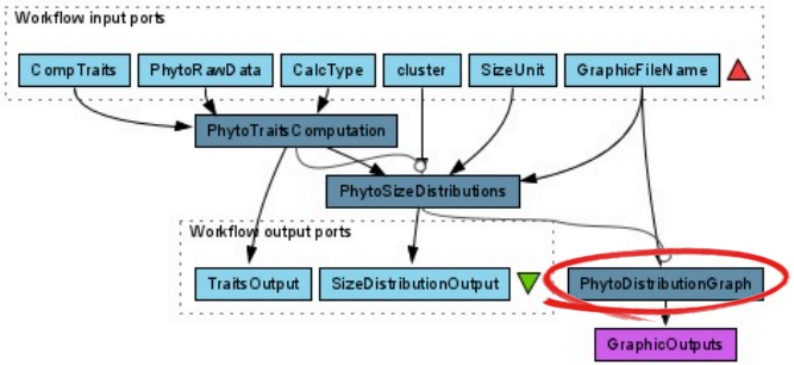
- SizeDistributionOutput: a csv file containing the same columns of the input csv file

ENVRI FAIR

Phytoplankton traits-based analysis SHOWCASE

STEPS

PhytoDistributionGraph: it defines the web path in which display results and gives the name to the pdf file



Input

- GraphicFileName: the name that will be used to create the pdf distribution file

Output

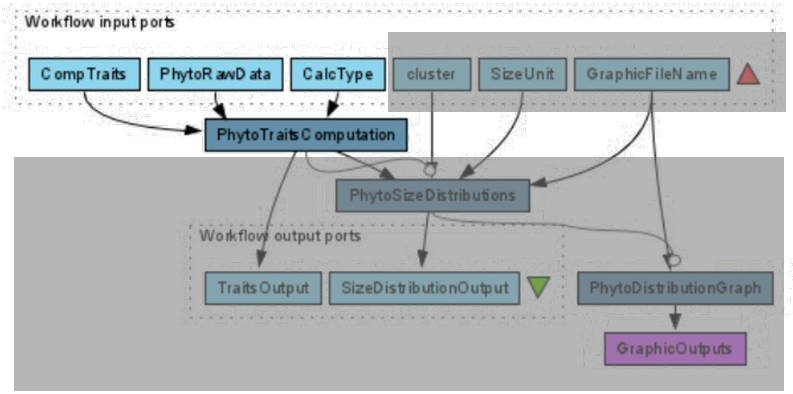
- GraphicOutput: the web page on the browser showing the graphs

Phytoplankton traits-based analysis SHOWCASE

EXECUTION

Inputs: let's suppose to give the following inputs

- CompTraits: BV → Biovolume
- PhytoRawData: PhytoplanktonWiserProject_input.csv
- CalcType: true → simplified computation type



ENVRI FAIR

Phytoplankton traits-based analysis SHOWCASE

EXECUTION

PhytoplanktonWiserProject_input.csv (64 columns, about 38K rows)

catalognumber	organismquantity	organismquantitytype	eventid	parenteventid	year	month	country	countrycode	locality	decimallatitude	decimallongitude	phylum	class	fa
20429	1	individual	bb	2	2009	9	italy	it	lesina	41.86828	15.36194	Chlorophyta	Chlorophyceae	
20469	1	individual	bb	2	2009	9	italy	it	lesina	41.86828	15.36194	Chlorophyta	Chlorophyceae	
20470	1	individual	bb	2	2009	9	italy	it	lesina	41.86828	15.36194	Chlorophyta	Chlorophyceae	
20471	1	individual	bb	2	2009	9	italy	it	lesina	41.86828	15.36194	Chlorophyta	Chlorophyceae	
20472	1	individual	bb	2	2009	9	italy	it	lesina	41.86828	15.36194	Chlorophyta	Chlorophyceae	
20473	1	individual	bb	2	2009	9	italy	it	lesina	41.86828	15.36194	Chlorophyta	Chlorophyceae	
20474	1	individual	bb	2	2009	9	italy	it	lesina	41.86828	15.36194	Chlorophyta	Chlorophyceae	
20601	1	individual	bb	2	2009	9	italy	it	lesina	41.86828	15.36194	Chlorophyta	Chlorophyceae	
20602	1	individual	bb	2	2009	9	italy	it	lesina	41.86828	15.36194	Chlorophyta	Chlorophyceae	
20657	1	individual	bb	2	2009	9	italy	it	lesina	41.86828	15.36194	Chlorophyta	Chlorophyceae	
20658	1	individual	bb	2	2009	9	italy	it	lesina	41.86828	15.36194	Chlorophyta	Chlorophyceae	
20684	1	individual	bb	2	2009	9	italy	it	lesina	41.86828	15.36194	Chlorophyta	Chlorophyceae	

!!! ***Mandatory fields***: scientific name, measurement remarks, length and width

ENVRI FAIR

Phytoplankton traits-based analysis SHOWCASE

EXECUTION

PhytoTraitsComputation

PhytoplanktonWiserProject_input.csv →  → TraitsOutput.csv

Phytoplankton traits-based analysis SHOWCASE

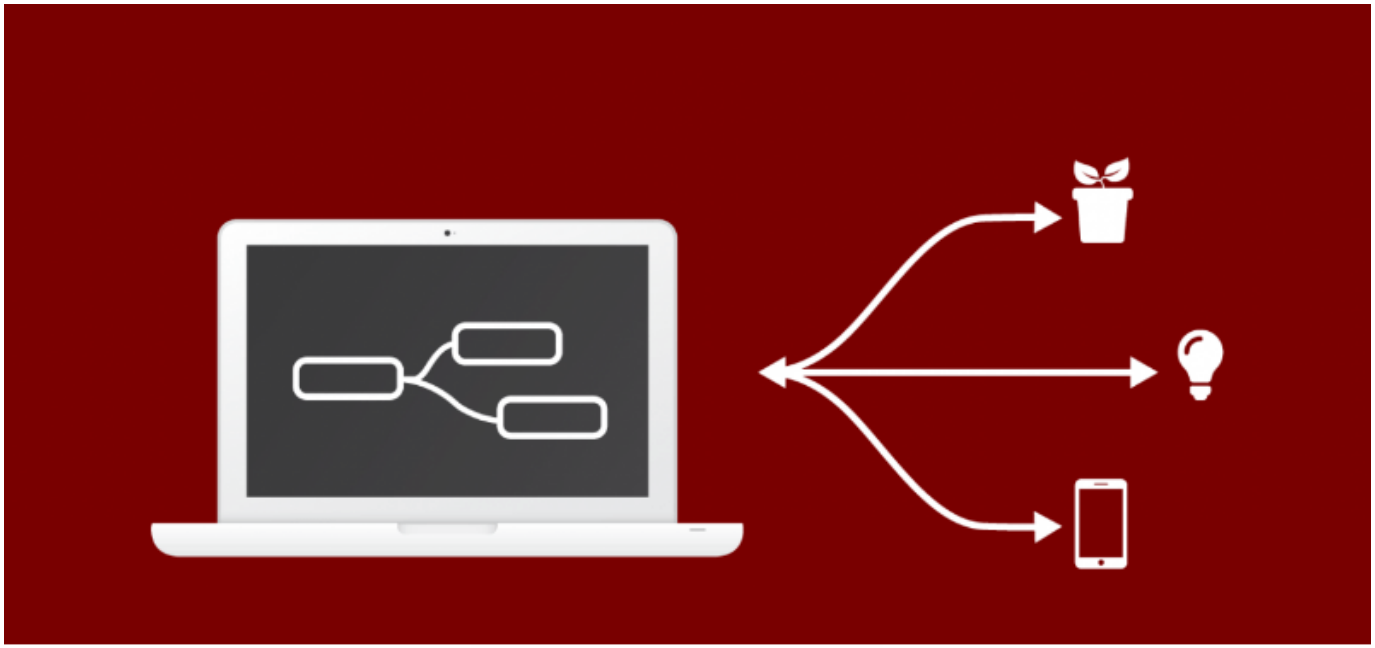
RESULTS

TraitsOutput.csv file

catalognumber	organismquantity	organismquantitytype	even	ures	shape	biovolume	cel	carboncontent	length	width	a	b	c	d	h
20429	1	individual	bb		prolate spheroid	98,14029333		13,42647695	6,98963	5,17842				5,17842	6,98963
20469	1	individual	bb		prolate spheroid	145,8045721		18,87193929	7,96129	5,91418				5,91418	7,96129
20470	1	individual	bb		prolate spheroid	145,8045721		18,87193929	7,96129	5,91418				5,91418	7,96129
20471	1	individual	bb		prolate spheroid	145,8045721		18,87193929	7,96129	5,91418				5,91418	7,96129
20472	1	individual	bb		prolate spheroid	62,23609771		9,075065585	6,01446	4,44553				4,44553	6,01446
20473	1	individual	bb		prolate spheroid	62,23609771		9,075065585	6,01446	4,44553				4,44553	6,01446
20474	1	individual	bb		prolate spheroid	62,23609771		9,075065585	6,01446	4,44553				4,44553	6,01446
20601	1	individual	bb		prolate spheroid	56,46548917		8,346546352	5,78337	4,31819				4,31819	5,78337
20602	1	individual	bb		prolate spheroid	56,46548917		8,346546352	5,78337	4,31819				4,31819	5,78337
20657	1	individual	bb		prolate spheroid	125,3508621		16,57153916	9,0703	5,13752				5,13752	9,0703
20658	1	individual	bb		prolate spheroid	125,3508621		16,57153916	9,0703	5,13752				5,13752	9,0703
20684	1	individual	bb		prolate spheroid	23,60155107		3,941869269	4,41684	3,19459				3,19459	4,41684
20685	1	individual	bb		prolate spheroid	23,60155107		3,941869269	4,41684	3,19459				3,19459	4,41684

.....

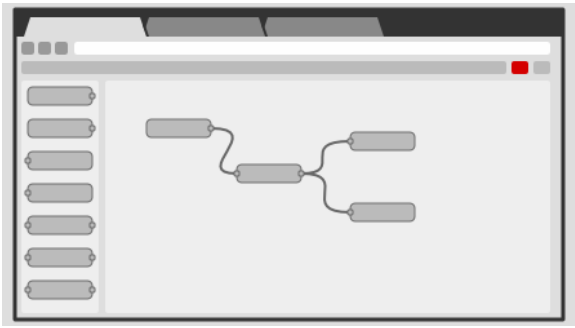
Node-RED



What is Node-RED?

https://nodered.org/

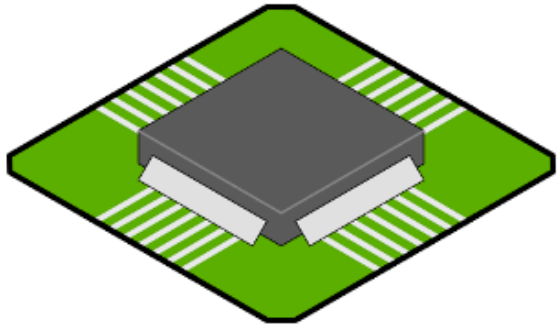
- A visual programming tool for wiring the Internet of Things developed by IBM Emerging Technology and the open source community.
- It provides a browser-based editor that allows users to easily wire up input, output and processing nodes in order to create flows able to process data, control things, or send alerts.
- Low-code programming for event-driven applications



- It is built on Node.js, taking full advantage of its event-driven, non-blocking model
- It has over 225.000 modules in its package repository → it is easy to extend the range of nodes to add new capabilities.

How can Node-RED be run?

Run locally



On a device

In the cloud

FRED (Front end for Node-RED)

https://fred.sensetecnic.com

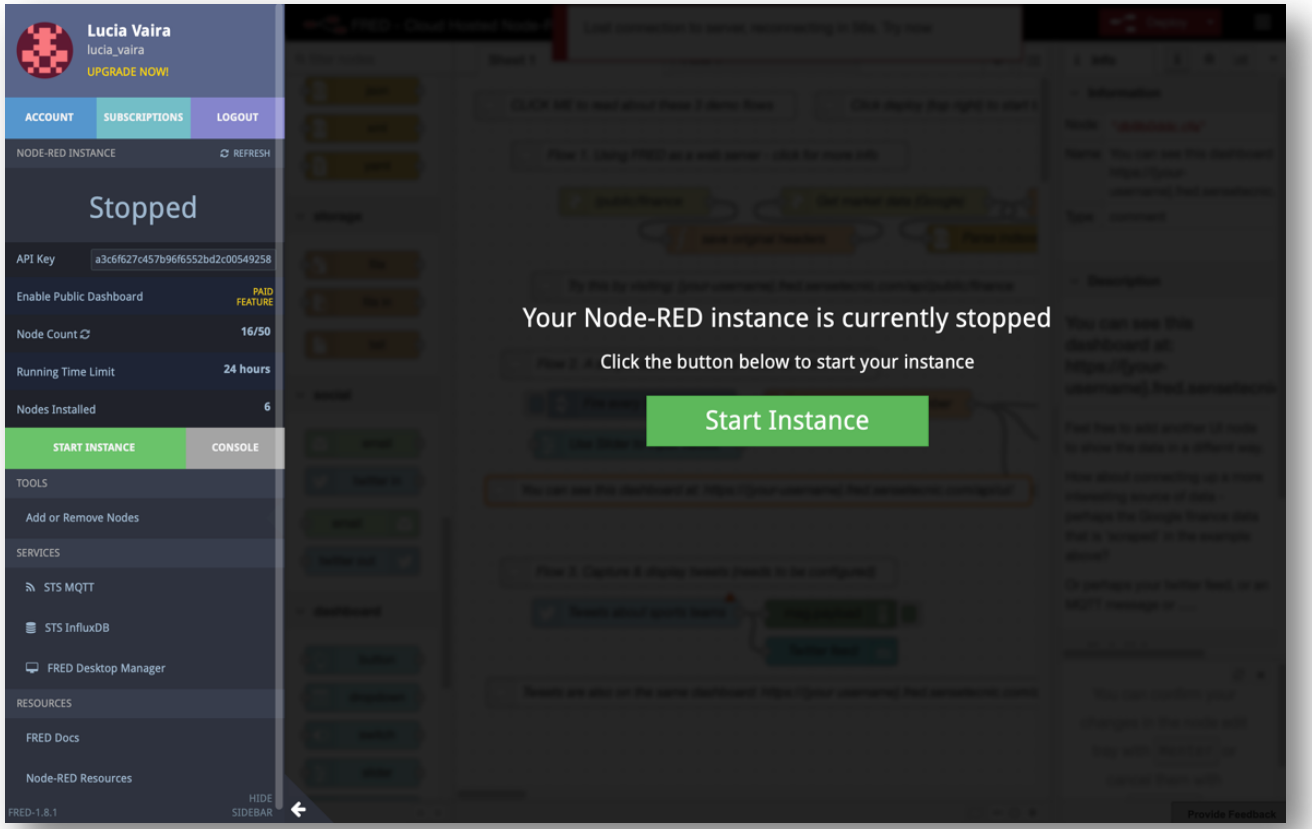
- FRED manages instances of Node-RED for multiple users in the cloud so that users have not to worry about accomplishing projects, setting up and maintaining the Node-RED instance.

*Read the "**Instructions for Participants**" that we have sent you by email yesterday and that are on the ENVRI Community Training Platform*

FRED FAQ: https://docs.sensetecnic.com/fred/faq/#q-whats-with-the-name

FRED (Front end for Node-RED)

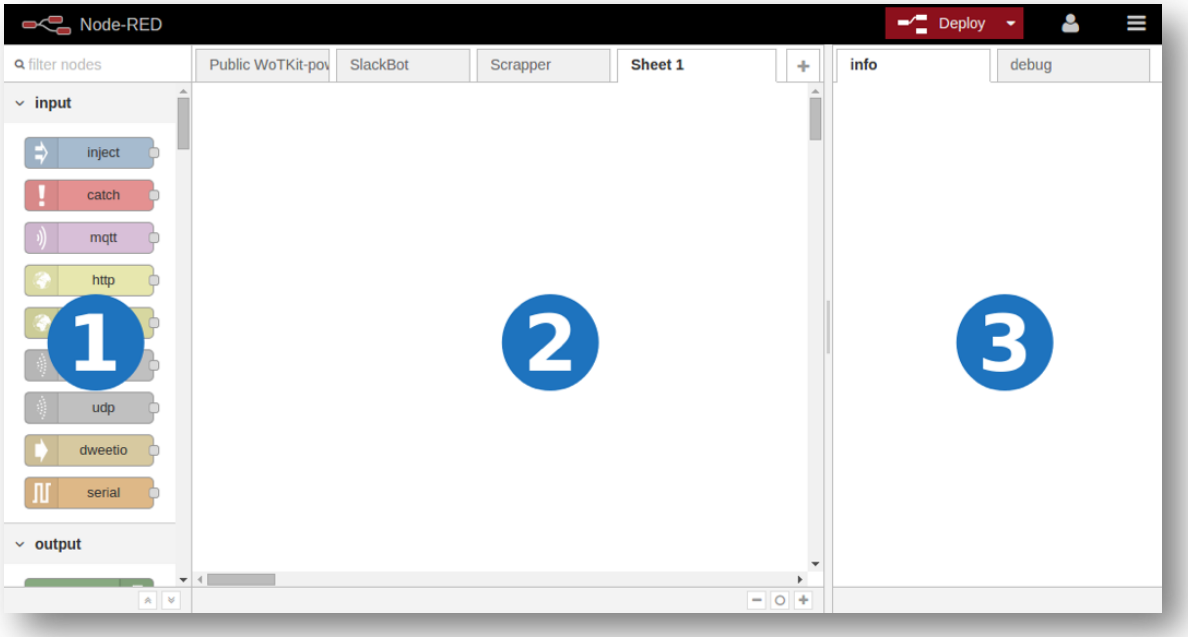
https://fred.sensetecnic.com



- Free account: fully functioning node-RED instance that you can use to build and run Node-RED flows on FRED cloud server.
- Three main restrictions:
  - max limit (50) on the number of nodes you can use in your flows
  - 24 **running** time limit
  - once deployed, your flows will run in FRED server for 24h

Node-RED

- Node-RED has three main components: Node Panel, Sheets Panel, Info and Debug Panel.

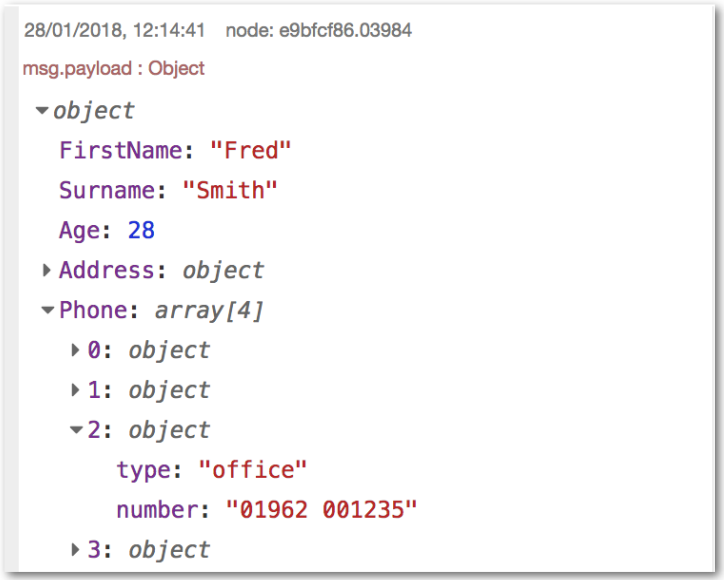


- 1) contains a list of nodes organized by categories (input, output, function, etc.)
- 2) is the working space in which drag and drops nodes
- 3) contains two tabs:
  - info providing info about a selected node
  - debug provides info printed to the debug console

ENVRI FAIR

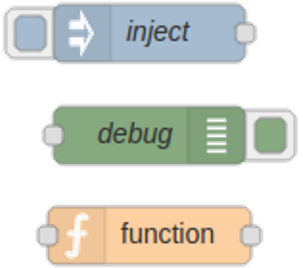
Node-RED

- Node-RED nodes consume ***input messages*** and produce ***output messages***.
- Messages are JSON (JavaScript Object Notation) objects that contain at least a payload parameter



Node-RED

There are three types of nodes:

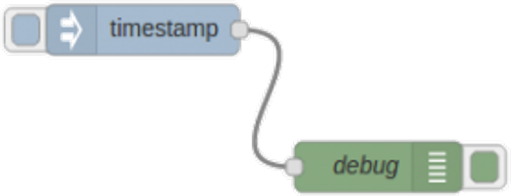


- **Input** Nodes (e.g. inject): have a white square only on their right side
- **Output** Nodes (e.g. debug): have a white square only on their left side
- **Processing** Nodes (e.g. function): have white squares on both left and right sides. They allow you to transform the data (e.g. json, csv, xml), use the data to trigger a message (e.g. trigger, delay), or to write custom code that uses the data received (e.g. function).

- Some nodes (e.g., Inject and debug) have a button that allows to actuate a node (in the case of the inject node) or to enable and disable a node (in the case of the debug node).

Node-RED

- **Nodes configuration**: double click on a node → form with node options
- **Flow creation**: draw a line between the white squares to the left and right of nodes we want to connect. The right white square represents a node's input; the left white square represents a node's output → in flows data moves from left to right.

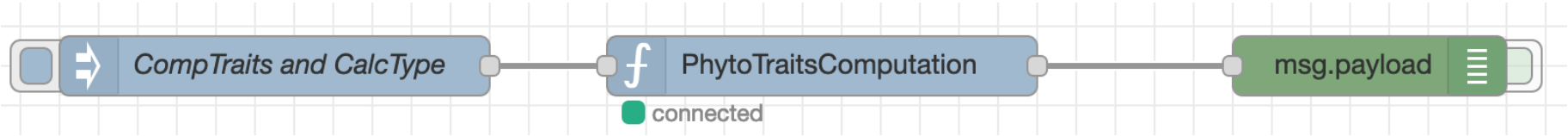


- The inject node (when the blue button on its left side is pressed) produces a timestamp that is consumed by the debug node and printed on the debug console.

Node-RED

- Once created a flow → deploy the flow clicking on the “Deploy” red button located on the right side of the top menu bar.
- Whatever change done on the flow, re-deploy is needed to save such change by clicking on the “Deploy” red button again.

Phyto VRE workflow on Node-RED

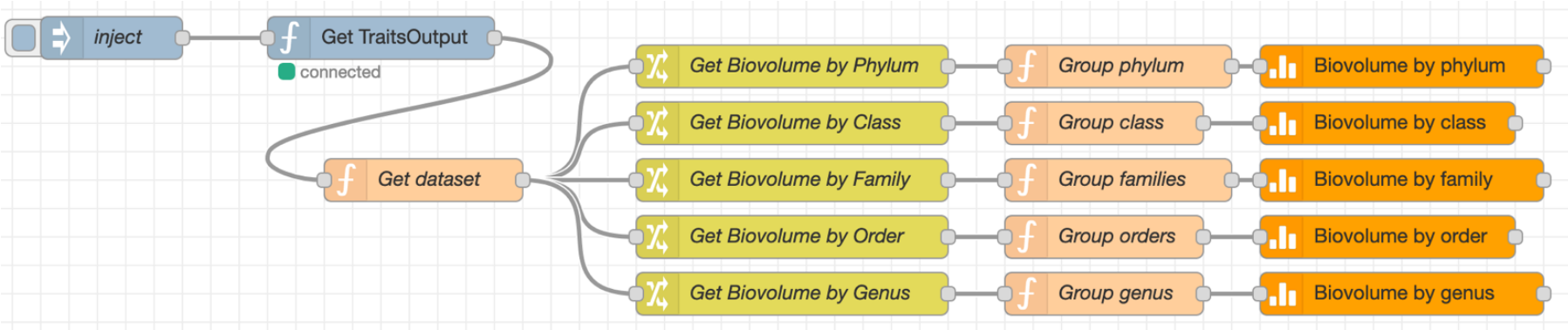


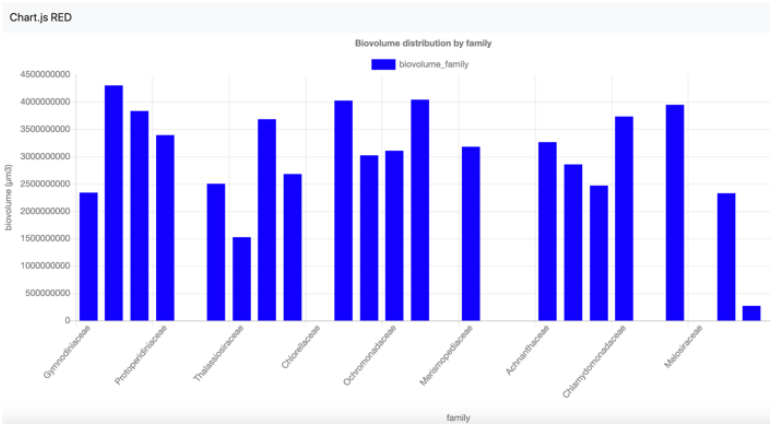
Inputs:
CompTraits: BV
CalcType: true

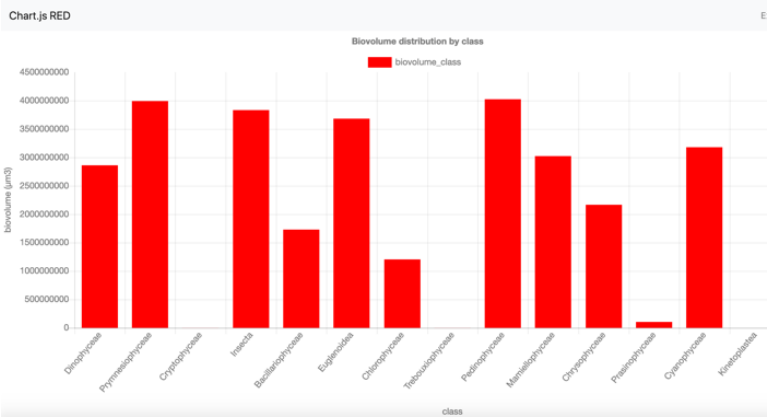
PhytoTraitsComputation:
it computes morphological and demographic traits: hidden dimension and biovolume

Output:
TraitsOutput.csv file that contains the biovolume for each sample

Phyto VRE workflow on Node-RED

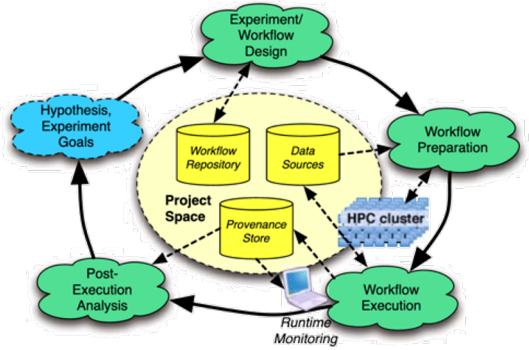






Phyto VRE workflow on Node-RED

- One of the strengths on Node-RED is its powerful and **user-friendly interface**
- Workflow Lifecycle: there are several stakeholders involved in the cycle



- If you have no ICT knowledge / support, there is also ***node-red-contrib-blockly,*** a Node Red node that offers a visual programming interface, to make programming a function node easier. Just drag and drop blocks to build your program logic, without having to write the JavaScript code yourself. By building your code in a visual way, you don't have to learn the JavaScript syntax, which makes programming very difficult for beginners. Moreover the generated JavaScript code can be displayed, so you can learn JavaScript coding step by step.
- https://flows.nodered.org/node/node-red-contrib-blockly

Examples on Node-RED

- Print the timestamp (once or at repeated intervals)
- Parse CSV input
- How to import/export flows in Node-RED
- Read from a local CSV file
- Write data to a local file
- Read from external resource
- Simple computations with R script
- Interface example

A quick poll

5. Which workflow could be interesting to you?

https://www.menti.com/

54 92 70

On a laptop: visit https://www.menti.com/ and put the code **549270**
On a smartphone: scan the QR code and answer

A quick poll

6. Do you have questions?

https://www.menti.com/

54 92 70

On a laptop: visit https://www.menti.com/ and put the code **549270**
On a smartphone: scan the QR code and answer