



Data management and cloud computing

Zhiming Zhao

University of Amsterdam,

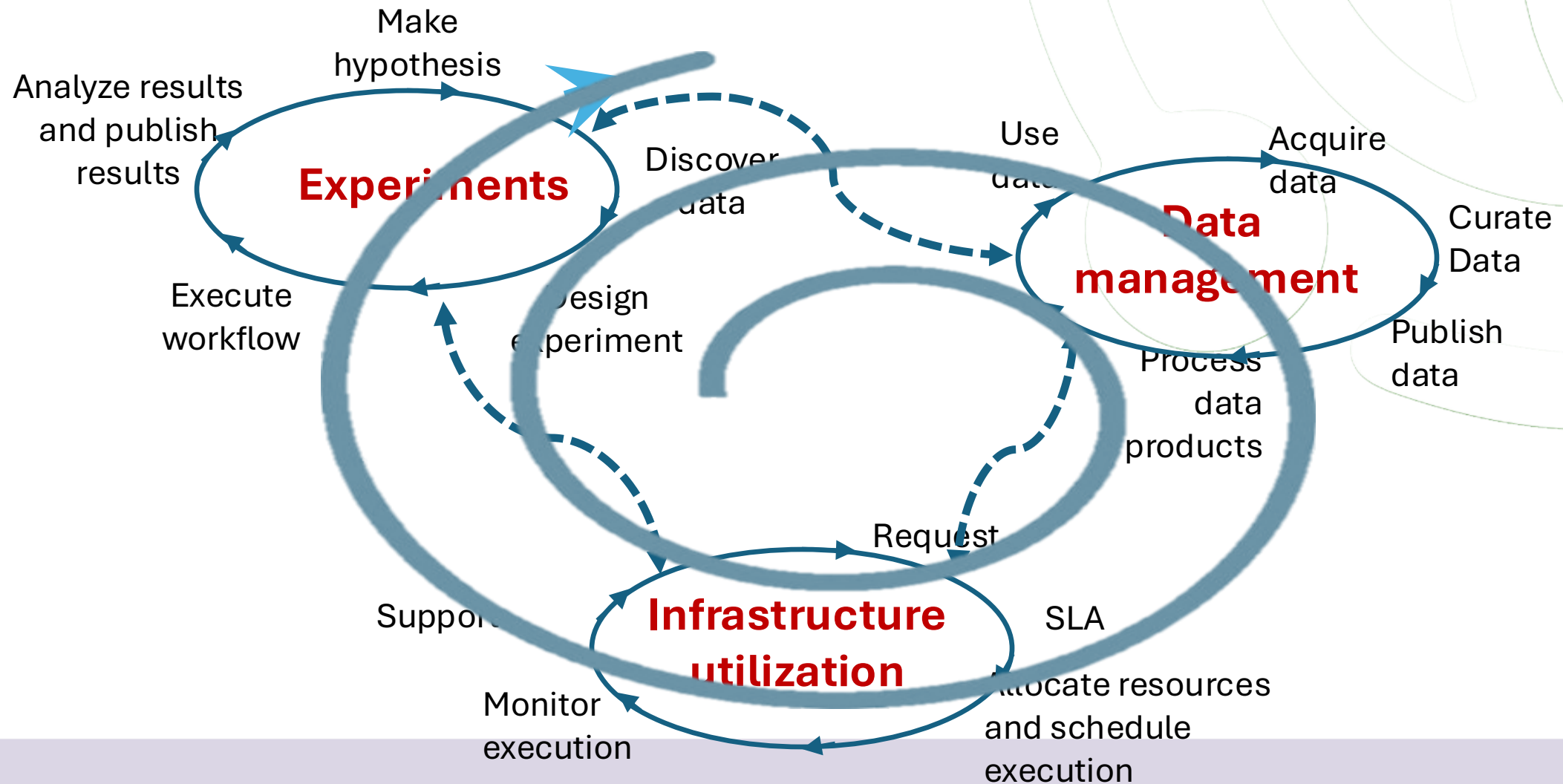
LifeWatch Virtual Lab & Innovation Center (VLIC)



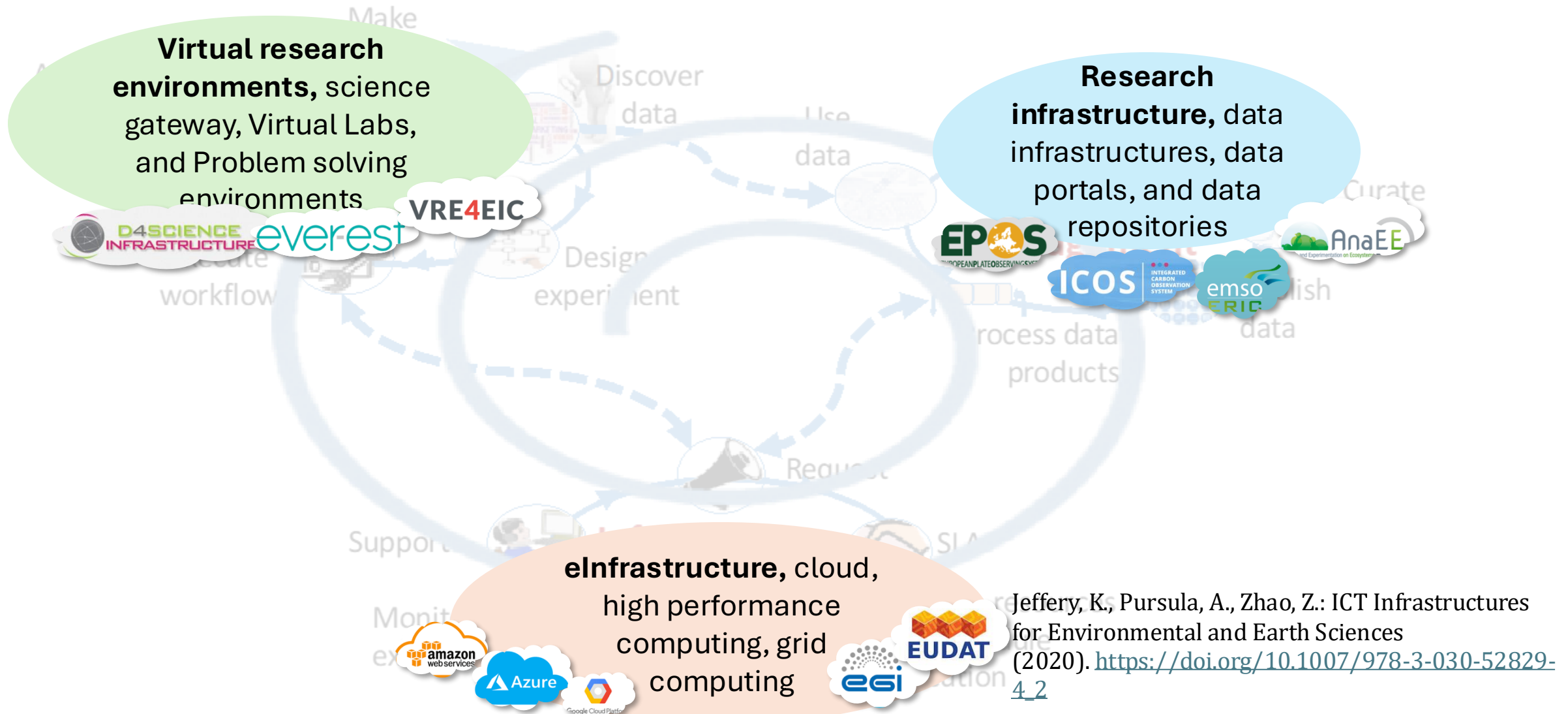
IR0000032 – ITINERIS, Italian Integrated Environmental Research Infrastructures System
(D.D. n. 130/2022 - CUP B53C22002150006) Funded by EU - Next Generation EU PNRR-
Mission 4 “Education and Research” - Component 2: “From research to business” - Investment
3.1: “Fund for the realisation of an integrated system of research and innovation infrastructures”



Research activities



Research support systems



Outline

- **Search** research assets
 1. Catalogue
 2. Search engine
- **Computing** and data processing
 3. High-performance and high-throughput computing
 4. Cloud computing concepts
- **Running applications in Cloud**
 5. **Service** and RESTful
 6. Workflow composition and automation
 7. Workflow **provenance**
- **Research data** management
 8. FAIRness
 9. Data quality control
- **Research software** quality
 10. Research software
 11. Research software quality assessment

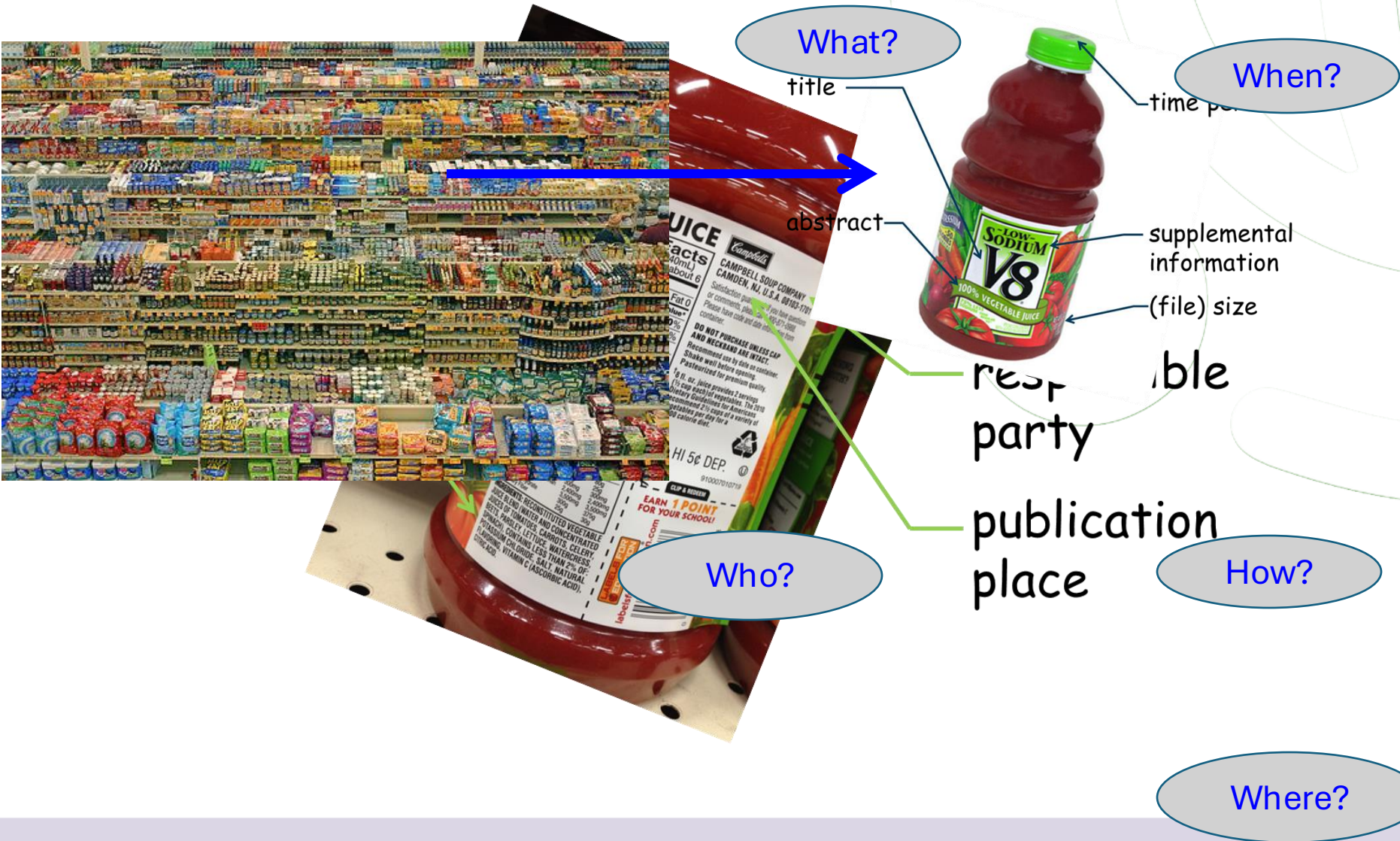


1. Metadata Catalogue

Discussion

 Have you used any data or service catalogues?

Metadata for product



What is a catalogue?

- 🌐 Collections in a Museum,
- 🌐 Products in shops,
- 🌐 Services in travel agency
- 🌐

Catalogue

Directory

Index

Inventory

portfolio

registry

list

....

....



← → ↻ 🏠 🔍 metadatacatalogue.lifewatch.eu/srv/eng/catalog.search;jsessionid=FD9E2B335FD210

📁 | ★ Bookmarks 🌐 Save to Mendeley 🚀 Getting Started 📁 Latest Headlines 📁 Imported From Fir... 🍏 An

LifeWatch
ERIC LifeWatch ERIC Metadata Catalogue 🔍 Search 🗺 Map

Search ...

← → ↻ 🏠 🔍 metadatacatalogue.lifewatch.eu/srv/eng

📁 | ★ Bookmarks 🌐 Save to Mendeley 🚀 Getting Started 📁


LifeWatch
ERIC LifeWatch ERIC Metadata Catalogue 🔍 Search

🔍 Back to search < Previous Next >


Steinbock (1937) Turbellaria XIV. The fishery grounds near Alexandria. Notes and Memories of the Fisheries Directorate of Egypt, No. 25

This is a historical dataset regarding Turbellaria species collected by professor Steurer; during his three month's stay in Alexandria. It contains large forms only, all of them being Polycladida. Some of them could not be well determined, as they were badly preserved and not yet mature.

About this resource

Categories	<div>  Datasets </div>	
Language	en	
Creator		
Creator		
Organization Name	Hellenic Centre for Marine Research	
Individual Name	First Name	Dimitra



Categories 

hery grounds near
eries Directorate of

ed by professor Steurer; during his
of them being Polycladida. Some of
and not yet mature.

Categories

d Landscape of the V
ological Reserve

the "Regional Protected Lands
Reserve (PPRLVCROM – "País
itológica de Mindelo"), a prot

OpenStreetMap contributors.

- simple example



How to make a catalogue?

Step 1: create metadata information of items

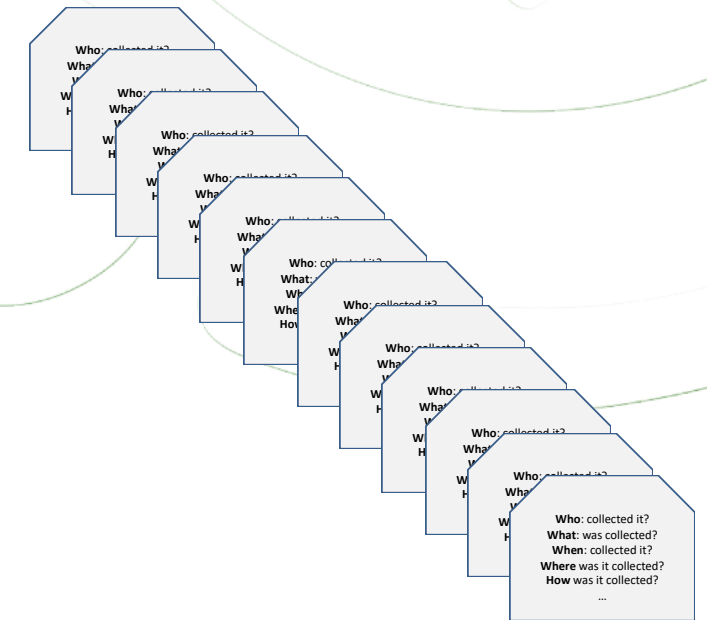


Who: collected it?
What: was collected?
When: collected it?
Where was it collected?
How was it collected?

...

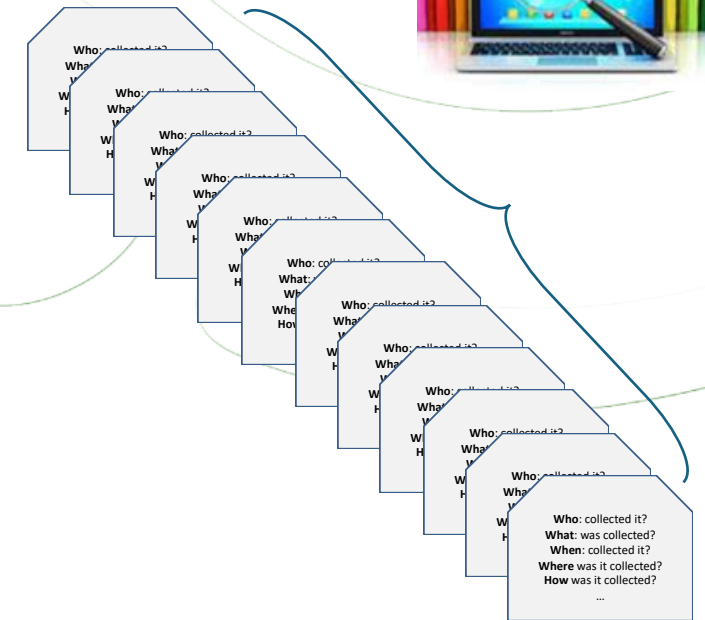
How to make a catalogue?

Step 2: organize the items



How to make a catalogue?

Step 3: provide interface for search



 The **Comprehensive Knowledge Archive Network (CKAN)** is

- a web-based open-source management system for the storage and distribution of open data,
- a powerful data catalogue system that is mainly used by public institutions seeking to share their data with the general public.

 Open source, Python web app, PostgreSQL DB, GPL

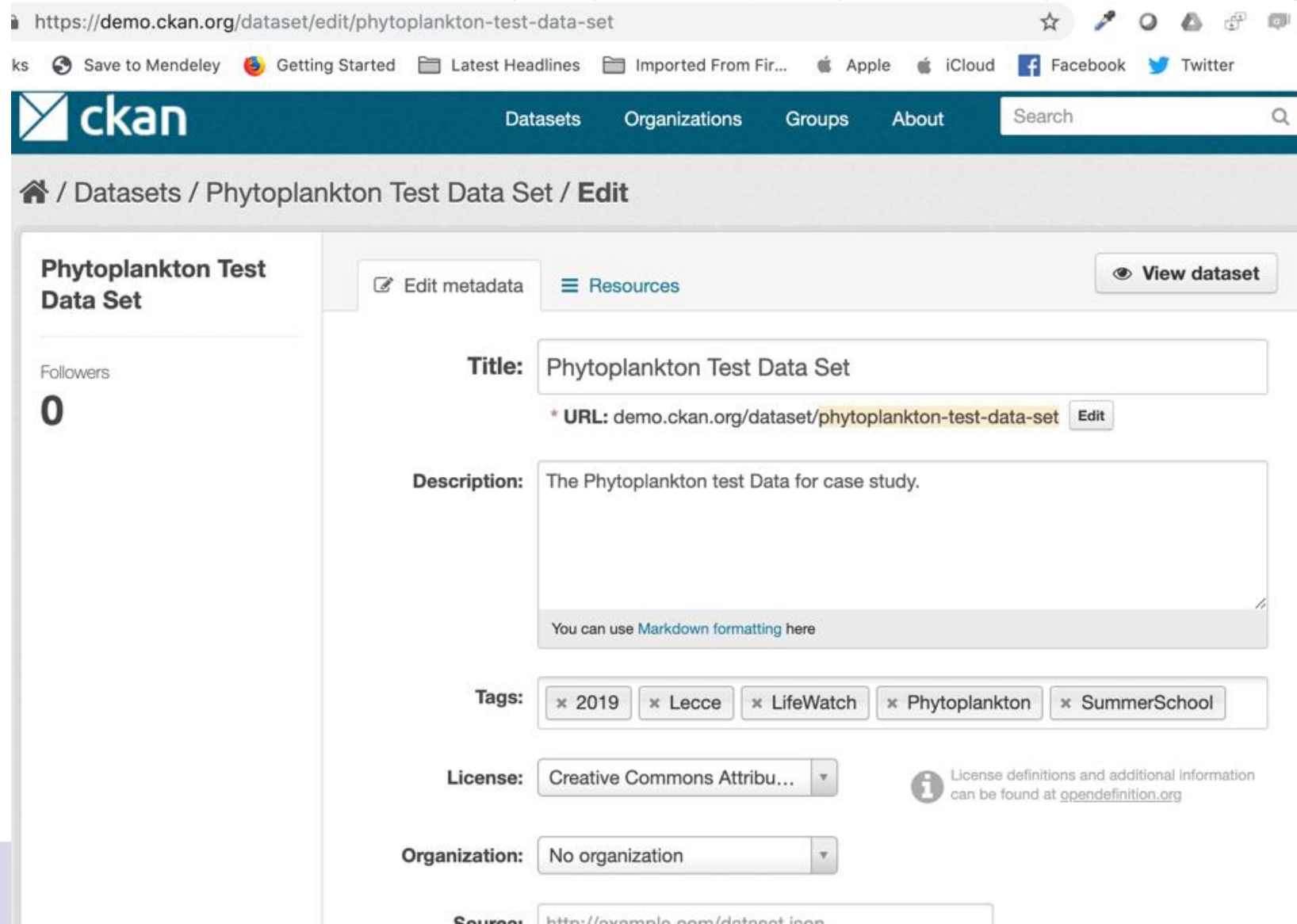
 <https://demo.ckan.org/>

Create data set

Edit metadata

Metadata

- Title
- Description
- Tag
- Organizations
- License
- Identifier



The screenshot shows the CKAN 'Edit metadata' interface for a dataset titled 'Phytoplankton Test Data Set'. The page includes a sidebar with the dataset name and a 'Followers' count of 0. The main content area has tabs for 'Edit metadata' (selected) and 'Resources'. A 'View dataset' button is in the top right. The form fields are as follows:


- Title:** Phytoplankton Test Data Set
- URL:** demo.ckan.org/dataset/phytoplankton-test-data-set (with an 'Edit' button)
- Description:** The Phytoplankton test Data for case study. (A note at the bottom says 'You can use [Markdown formatting](#) here')
- Tags:** 2019, Lecce, LifeWatch, Phytoplankton, SummerSchool
- License:** Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International (with a link to [opendefinition.org](#) for more info)
- Organization:** No organization
- Source:** http://example.com/dataset-json

Metadata in CKAN

- 🌐 **Title** – allows intuitive labelling of the dataset for search, sharing and linking.
- 🌐 **Unique identifier** – dataset has a unique URL which is customizable by the publisher.
- 🌐 **Groups** – display of which groups the dataset belongs to if applicable. Groups (such as science data) allow easier data linking, finding and sharing amongst interested publishers and users.
- 🌐 **Description** – additional information describing or analysing the data. This can either be static or an editable wiki which anyone can contribute to instantly or via admin moderation.
- 🌐 **Data preview** – preview .csv data quickly and easily in browser to see if this is the dataset you want.
- 🌐 **Revision history** – CKAN allows you to display a revision history for datasets which are freely editable by users (as is thedatahub.org)
- 🌐 **Extra fields** – these hold any additional information, such as location data (see geospatial feature) or types relevant to the publisher or dataset. How and where extra fields display is customizable.
- 🌐 **Licence** – instant view of whether the data is available under an open licence or not. This makes it clear to users whether they have the rights to use, change and re-distribute the data.
- 🌐 **Tags** – see what labels the dataset in question belongs to. Tags also allow for browsing between similarly tagged datasets in addition to enabling better discoverability through tag search and faceting by tags.
- 🌐 **Multiple formats (if provided)** – see the different formats the data has been made available in quickly in a table, with any further information relating to specific files provided inline.
- 🌐 **API key** – allows access every metadata field of the dataset and ability to change the data if you have the relevant permissions via API.

Other metadata standards related to catalogues

 Dublin CORE

 ISO 19115

 CKAN

 DCAT

 CERIF

Technologies



- 🌐 The GeoNetwork project started out in year 2001 as a **Spatial Data Catalogue System** for the Food and Agriculture organisation of the United Nations (**FAO**), the United Nations World Food Programme (**WFP**) and the United Nations Environmental Programme (**UNEP**).
- 🌐 At present the project is widely used as the basis of **Spatial Data Infrastructures** all around the world.
- 🌐 The project is part of the **Open Source Geospatial Foundation (OSGeo)** and can be found at GeoNetwork opensource.



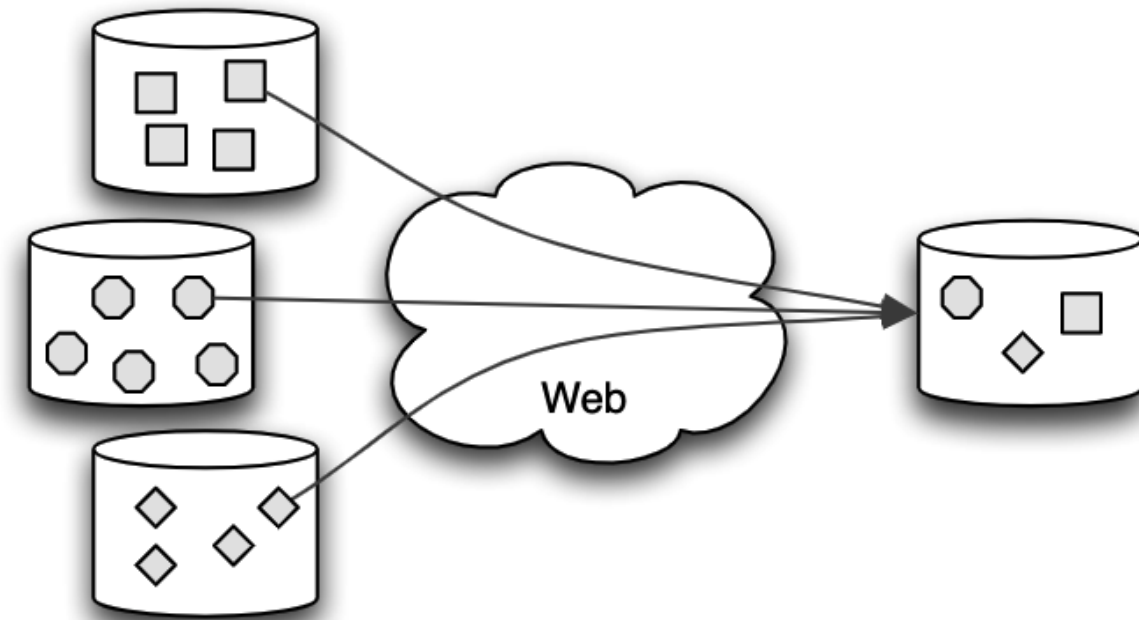
Discussion

 How does a catalogue enable search?

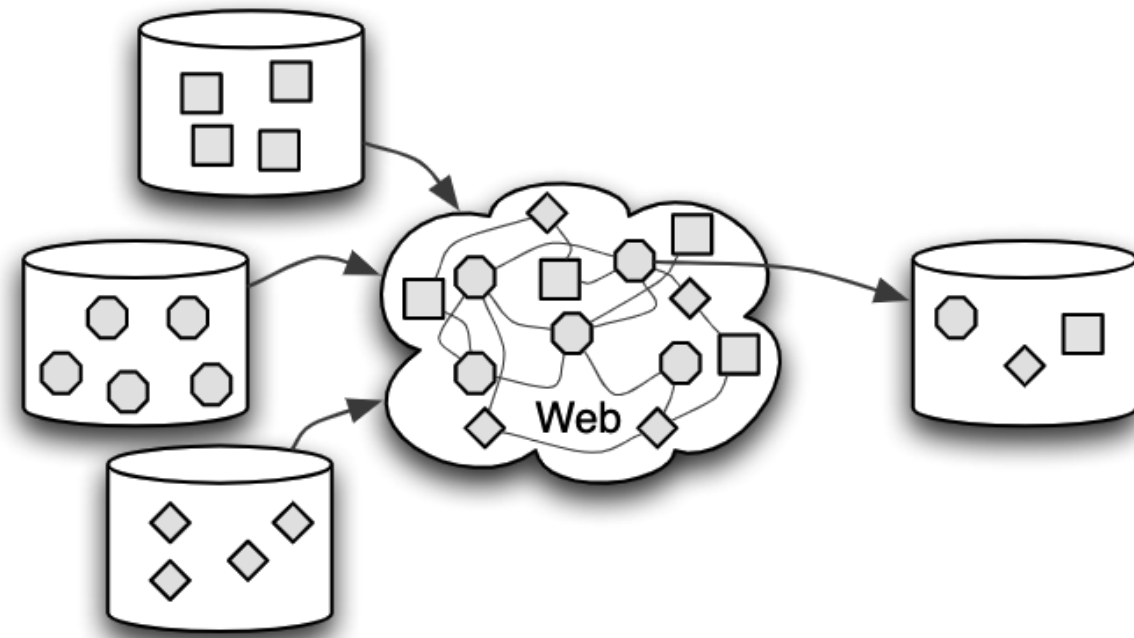
Other solutions

 OAI-PMH (Open Archives Initiative Protocol for Metadata Harvesting)

 Linked Open Data approach

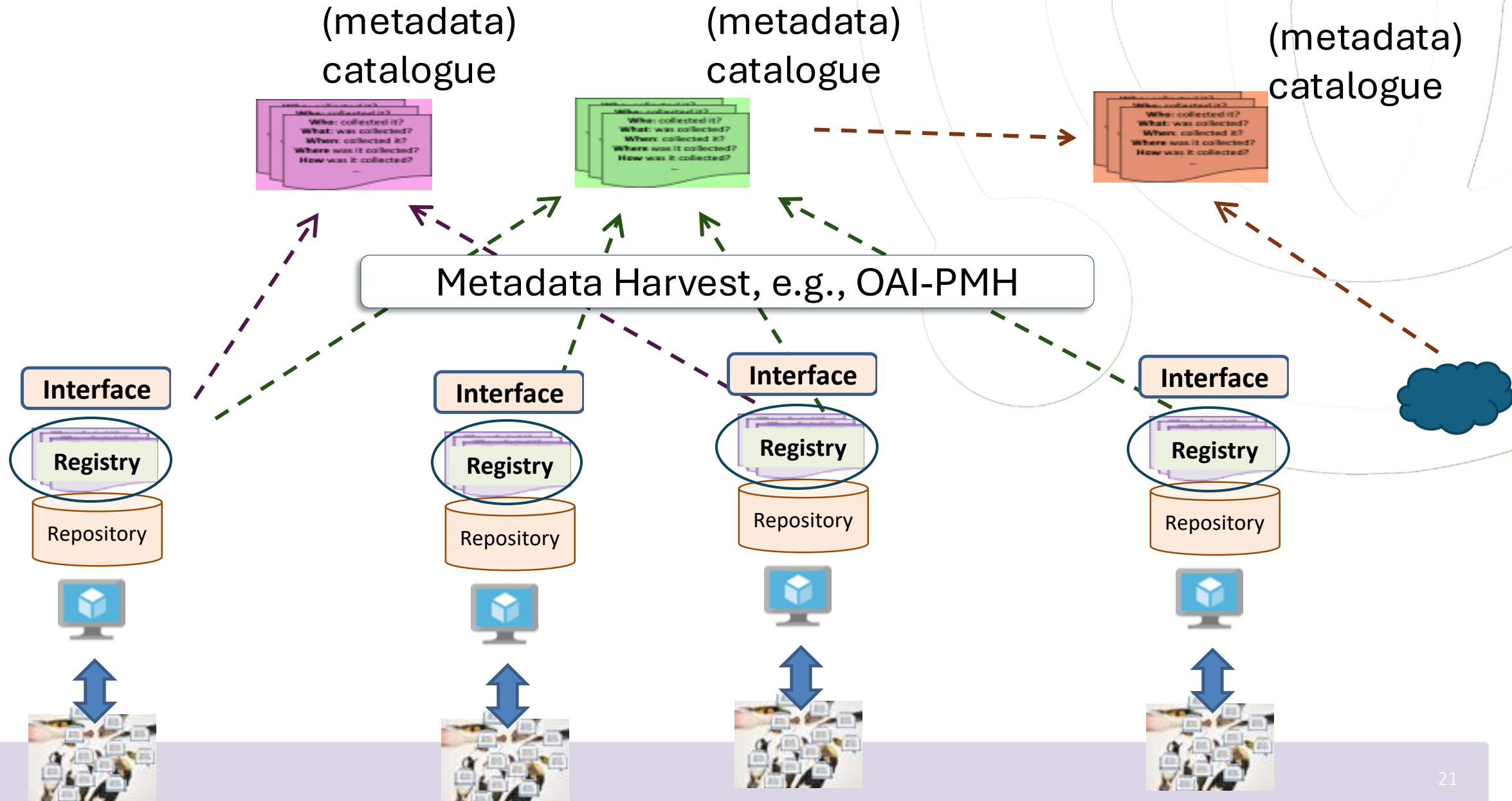


OAI-PMH Approach



LOD Approach

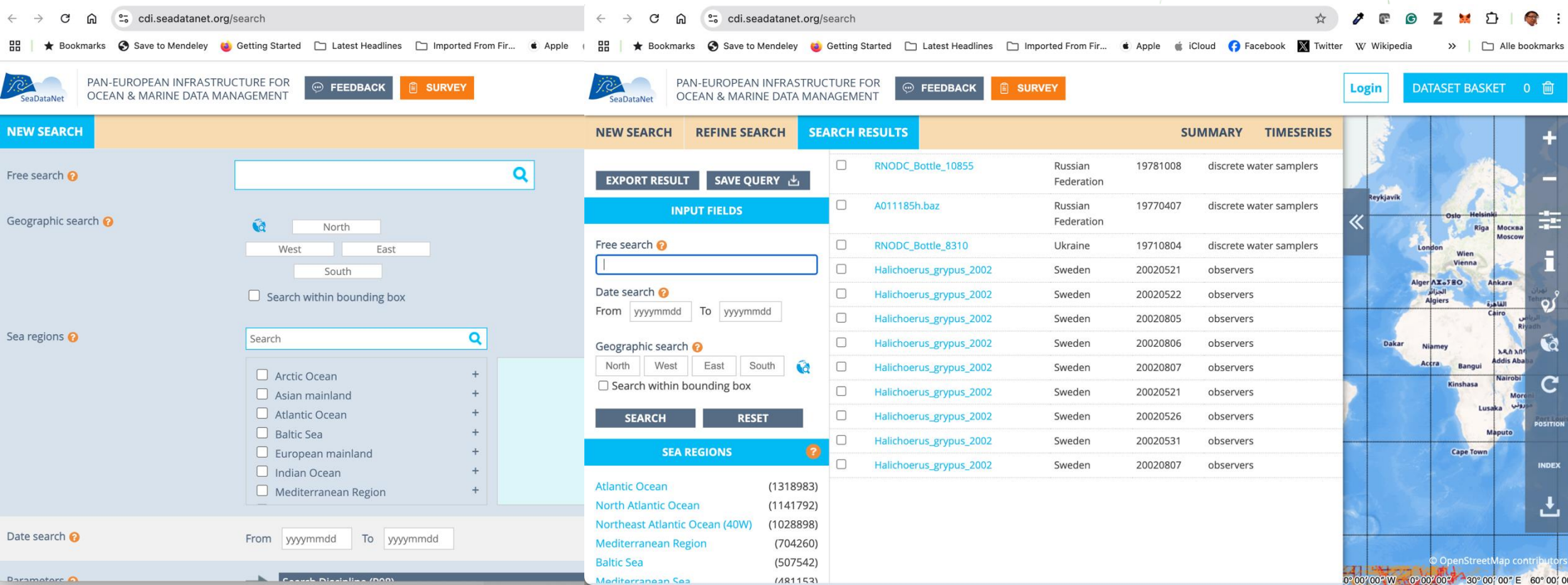
Catalogue for publishing and discovery



Discussion

How does a catalogue enable search?

- Keyword – free keyword, or based on metadata
- Filtering (based on Facets)



The screenshot displays the SeaDataNet search interface. The top navigation bar includes the SeaDataNet logo, the text "PAN-EUROPEAN INFRASTRUCTURE FOR OCEAN & MARINE DATA MANAGEMENT", and buttons for "FEEDBACK" and "SURVEY". A "Login" button and a "DATASET BASKET" with a count of 0 are also present.

The main interface is divided into several sections:

- NEW SEARCH:** Contains a "Free search" input field, a "Geographic search" section with a map and bounding box options, and a "Sea regions" section with a list of regions and their counts.
- REFINE SEARCH:** Includes an "EXPORT RESULT" button, a "SAVE QUERY" button, and an "INPUT FIELDS" section with a "Free search" input field, a "Date search" section, and a "Geographic search" section.
- SEARCH RESULTS:** A table displaying search results with columns for checkboxes, sample IDs, countries, dates, and descriptions.
- SUMMARY:** A section for summarizing the search results.
- TIMESERIES:** A section for viewing time series data.
- Map:** A map of Europe and Africa showing the location of the search results.

The "Sea regions" section lists the following regions and their counts:

Region	Count
Arctic Ocean	1318983
Asian mainland	1141792
Atlantic Ocean	1028898
Baltic Sea	704260
European mainland	507542
Indian Ocean	481153
Mediterranean Region	

The "SEARCH RESULTS" table shows the following data:

Sample ID	Country	Date	Description
RNODC_Bottle_10855	Russian Federation	19781008	discrete water samplers
A011185h.baz	Russian Federation	19770407	discrete water samplers
RNODC_Bottle_8310	Ukraine	19710804	discrete water samplers
Halichoerus_grypus_2002	Sweden	20020521	observers
Halichoerus_grypus_2002	Sweden	20020522	observers
Halichoerus_grypus_2002	Sweden	20020805	observers
Halichoerus_grypus_2002	Sweden	20020806	observers
Halichoerus_grypus_2002	Sweden	20020807	observers
Halichoerus_grypus_2002	Sweden	20020521	observers
Halichoerus_grypus_2002	Sweden	20020526	observers
Halichoerus_grypus_2002	Sweden	20020531	observers
Halichoerus_grypus_2002	Sweden	20020807	observers

Discussion

- 🌐 How does a catalogue enable search?
- 🌐 How can I find the most suitable one?

2. Search engine

Discussion

 What is your search experience?

How does a search engine work?

 Index documents based on their keywords

 Search queries from the index database

- Compute the similarity between documents and queries
- Rank the similarity among selected documents and present them to the user



“bag of words” in text search

- if search {“biodiversity”, “digital”, “twin”} from different documents
 - Document 1: contains (“*essential*”, “biodiversity”, “*variables*”)
 - Document 2: contains (“digital”, “samples”)
 - Document 3: contains (“physical”, “twin”)
- Assumption
 - A document might be relevant if it contains one of the keywords
 - A document might be more relevant if it contains a keyword many times
 - If a short document contains a keyword once , it might be more relevant than another long document that contains that keyword once

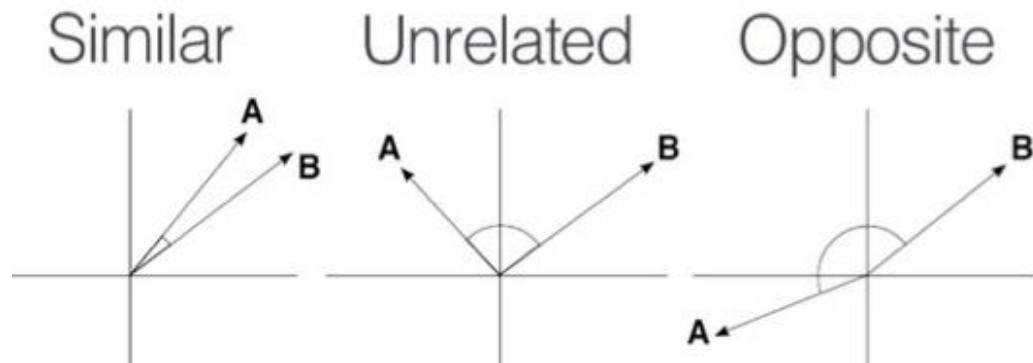
Basic idea of document similarity: vector space

 Represent documents and possible queries as an N-dimensional vector space;

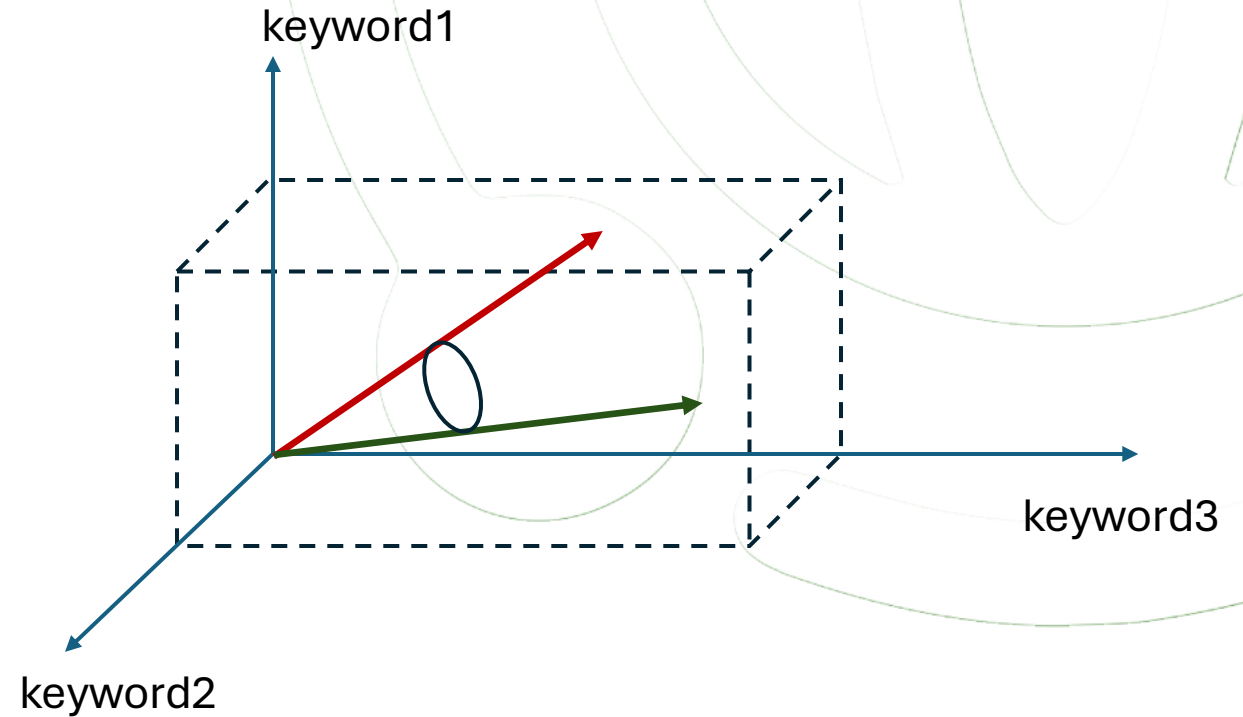
- Term: basic concepts and words in all documents and queries.
- Each term defines a dimension in the vector
- Document vector: (t_1, t_2, \dots, t_n) ; t_i is document term weight
- Query vector: (q_1, q_2, \dots, q_n) ; q_i is query term weight
- Relevance (q, d)

Similarity

Cosine similarity between vectors

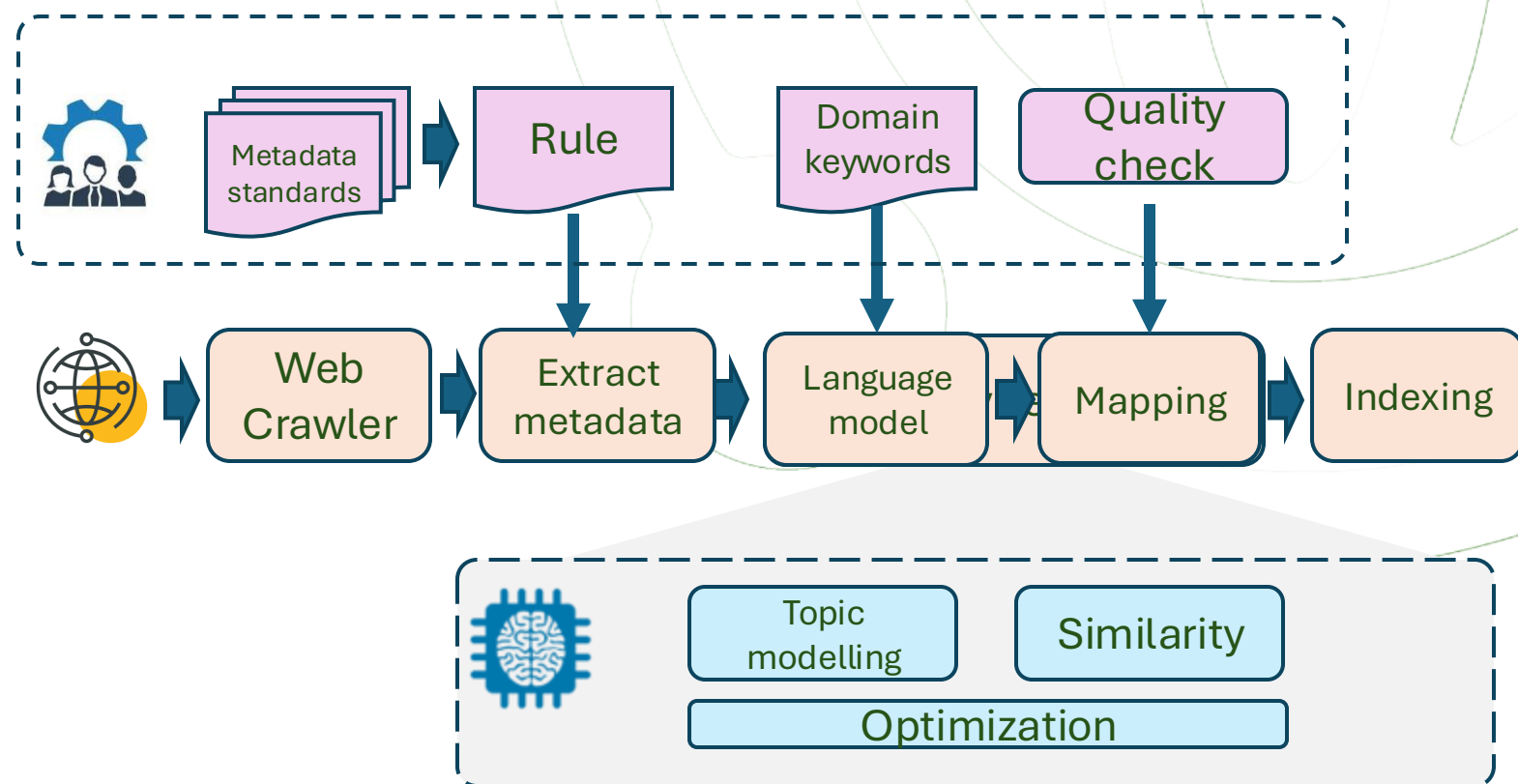


$$\cos \theta = \frac{\vec{a} \cdot \vec{b}}{\|\vec{a}\| \cdot \|\vec{b}\|}$$

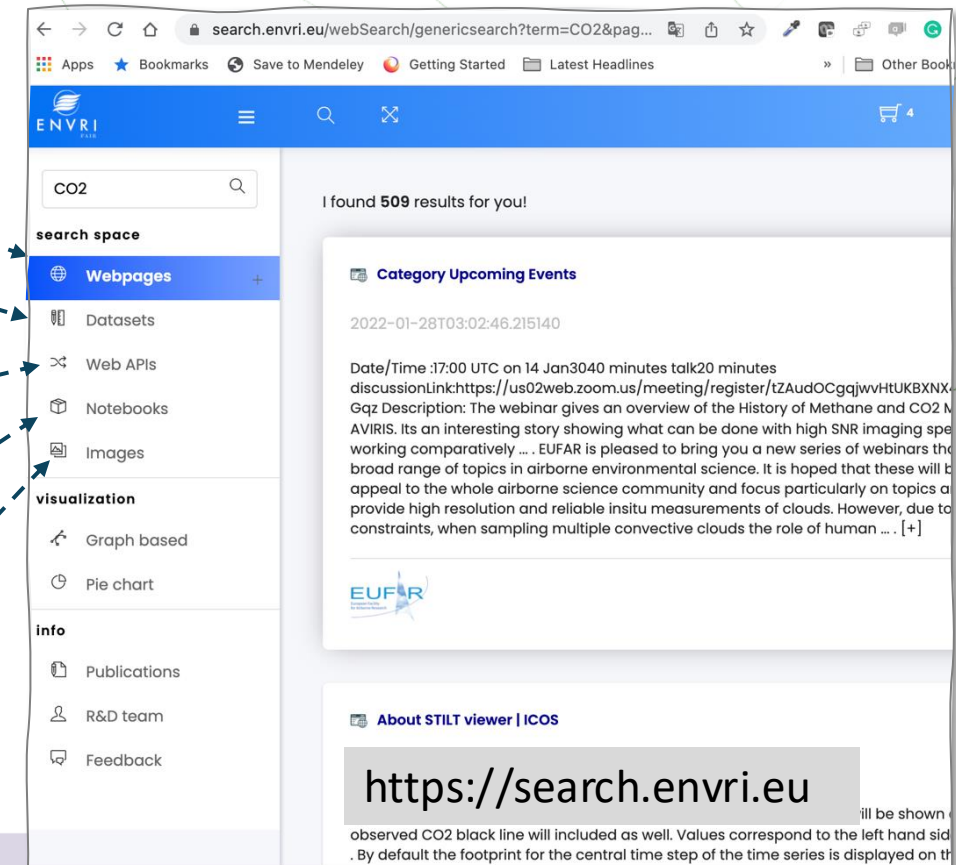
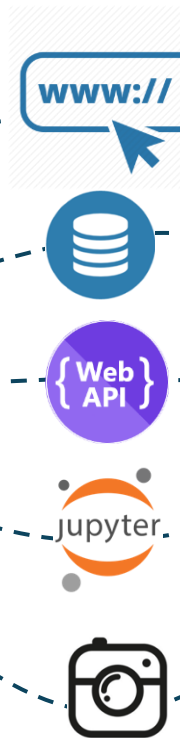
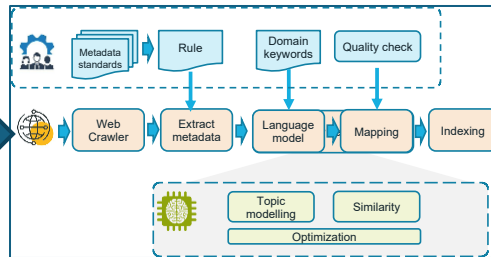


A customizable online information index framework

How to index metadata of research assets?

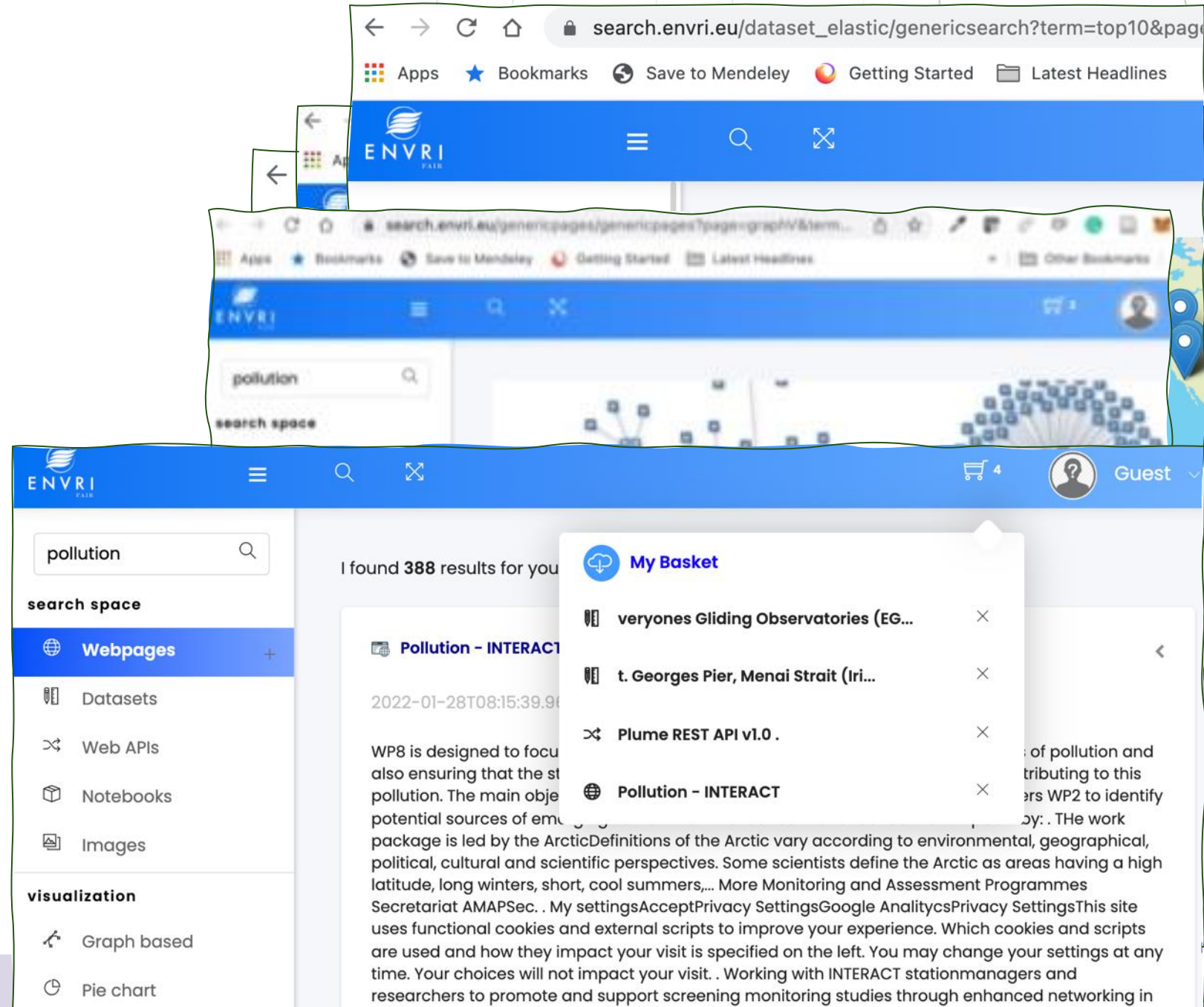


Search engine for different resource types



ENVRI Knowledgebase

- Knowledge base interface adopts the design style of ENVRI-HUB
- Category online resources as: web pages, data sets, Notebook, API, images, (more to come)
- Content filtering
- Better visualization
- Concept of basket

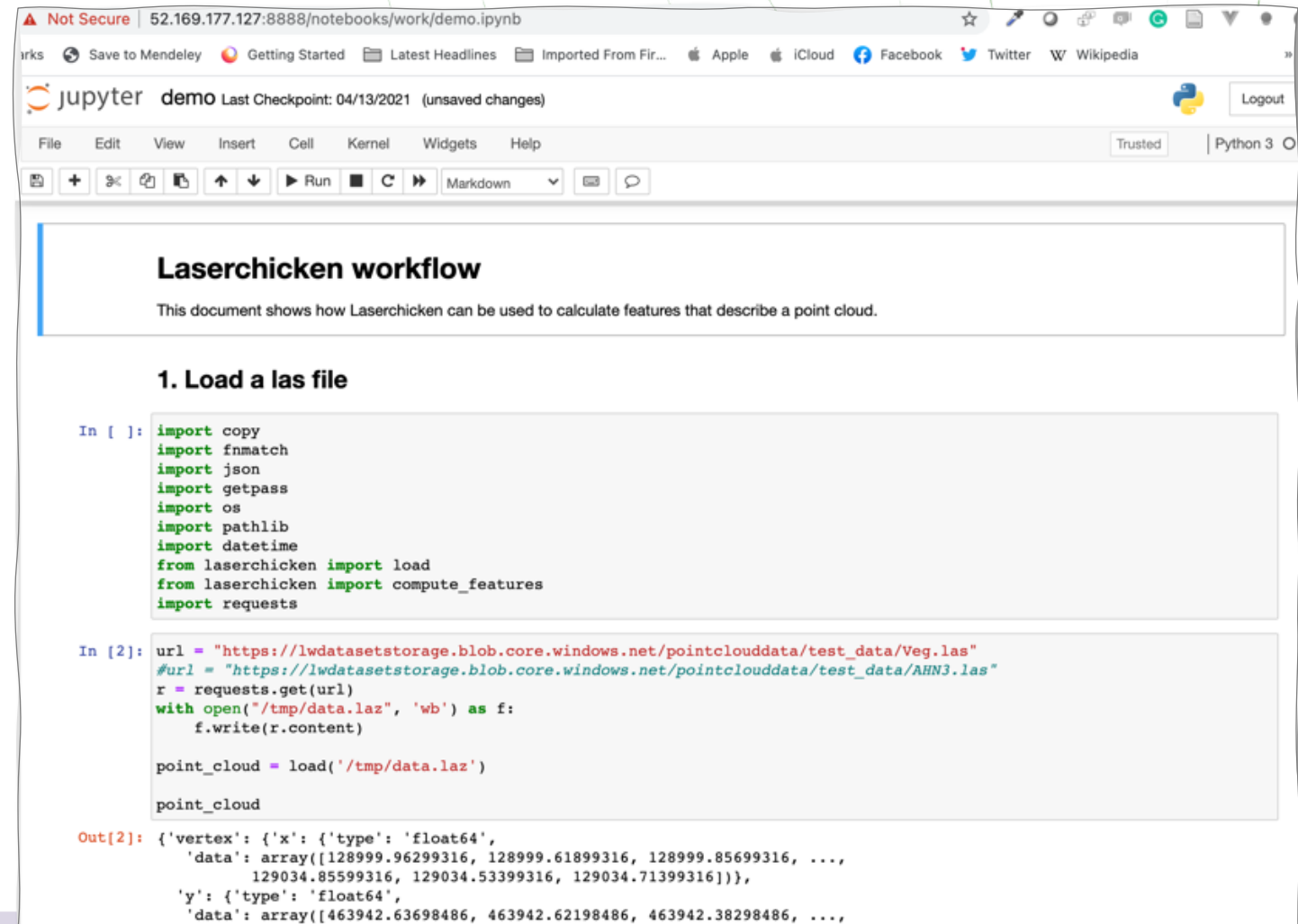


Search jupyter notebook

 Index the web information of the notebook from GitHub.

 Ongoing actions:

- Index the text part (cells) of notebooks
- Index the code patterns from the notebook (e.g., AI pipeline, models etc.)



The screenshot shows a Jupyter Notebook interface in a web browser. The browser address bar shows the URL `52.169.177.127:8888/notebooks/work/demo.ipynb`. The Jupyter Notebook header includes the Jupyter logo, the word "demo", and a "Last Checkpoint: 04/13/2021 (unsaved changes)" message. The notebook has a menu bar with "File", "Edit", "View", "Insert", "Cell", "Kernel", "Widgets", and "Help". Below the menu bar is a toolbar with icons for saving, opening, and running cells. The notebook content is titled "Laserchicken workflow" and includes a description: "This document shows how Laserchicken can be used to calculate features that describe a point cloud." The first section is "1. Load a las file". It contains two code cells. The first code cell is labeled "In []:" and contains the following code:

```
import copy
import fnmatch
import json
import getpass
import os
import pathlib
import datetime
from laserchicken import load
from laserchicken import compute_features
import requests
```

 The second code cell is labeled "In [2]:" and contains the following code:

```
url = "https://lwdatasetstorage.blob.core.windows.net/pointclouddata/test_data/Veg.las"
#url = "https://lwdatasetstorage.blob.core.windows.net/pointclouddata/test_data/AHN3.las"
r = requests.get(url)
with open("/tmp/data.laz", 'wb') as f:
    f.write(r.content)




point_cloud = load('/tmp/data.laz')

point_cloud
```

 The output of the second code cell is labeled "Out[2]:" and shows a dictionary with the following structure:

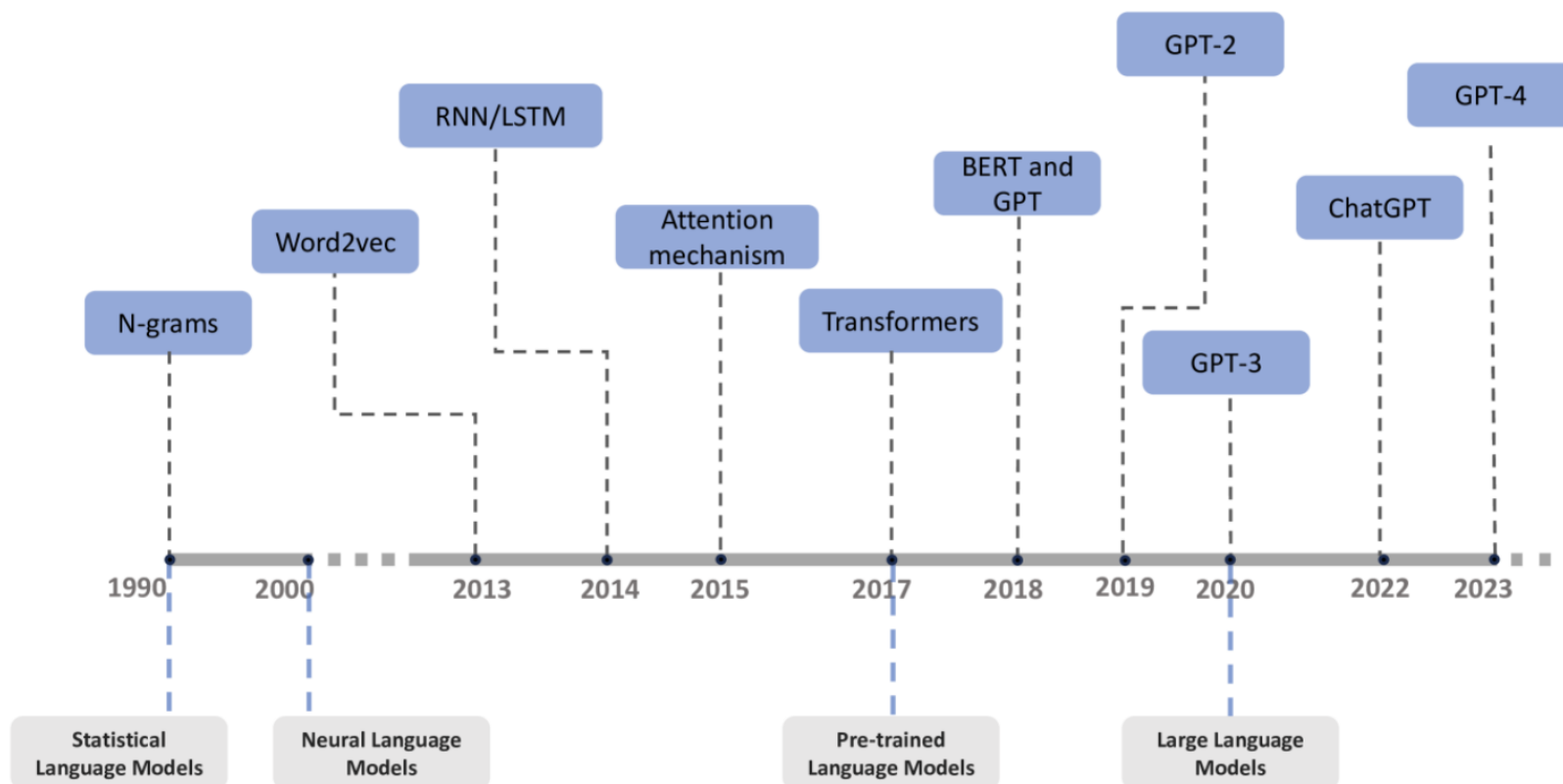
```
{
  'vertex': {
    'x': {
      'type': 'float64',
      'data': array([128999.96299316, 128999.61899316, 128999.85699316, ...,
                    129034.85599316, 129034.53399316, 129034.71399316])
    },
    'y': {
      'type': 'float64',
      'data': array([463942.63698486, 463942.62198486, 463942.38298486, ...,
                    ...])
    }
  }
}
```


Discussion: how to improve the search quality?

-  How to better rank the relevance?
-  How to “guess” the intent of the user?
-  ...

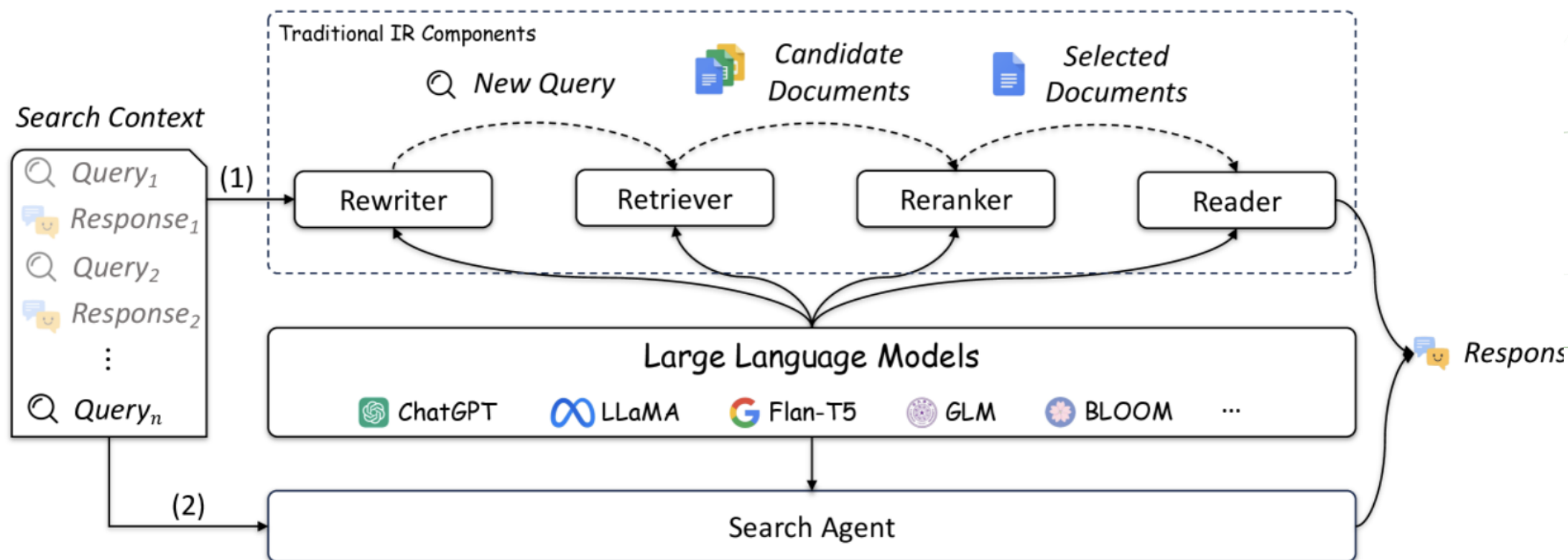
Discussion: Search using Advanced AI

🌐 Large language model (LLM)



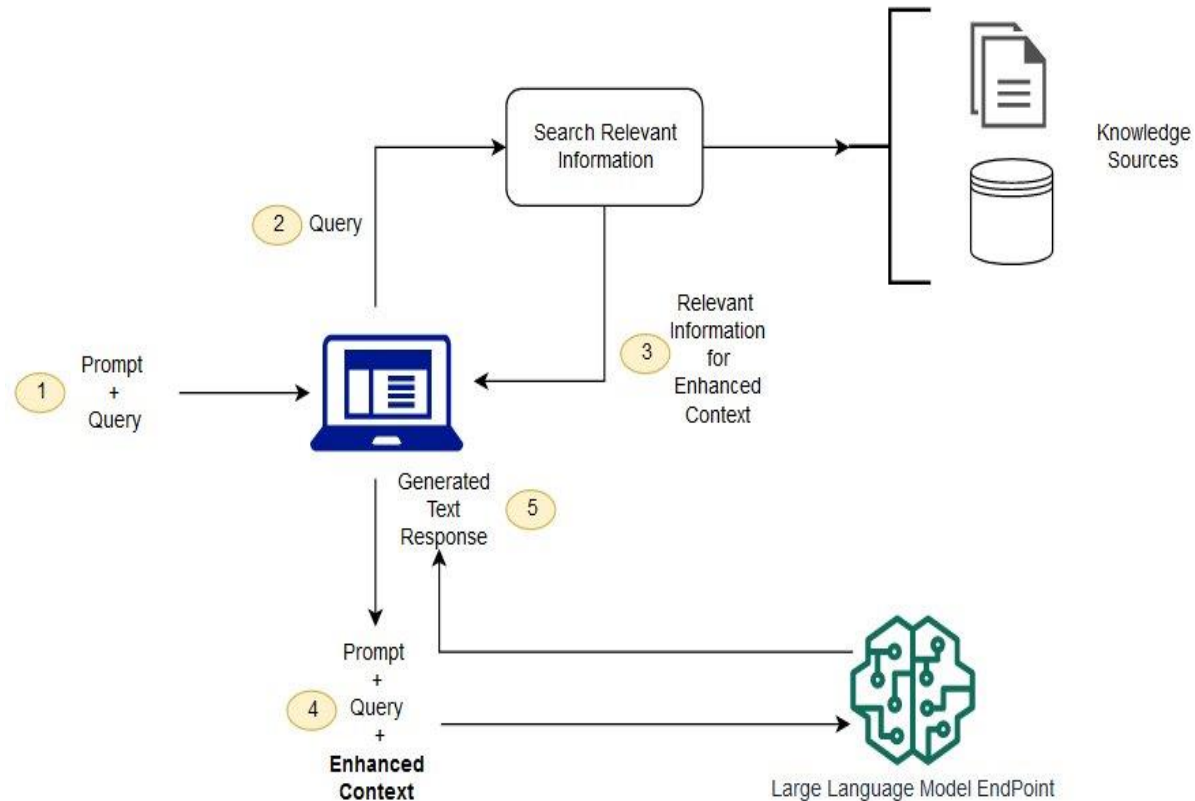
Discussion: LLM in search

LLM in different search phases



Discussion: LLM in search

Retrieval – Augmented Generation (RAG)



<https://aws.amazon.com/what-is/retrieval-augmented-generation/>

Discussion: Semantic search

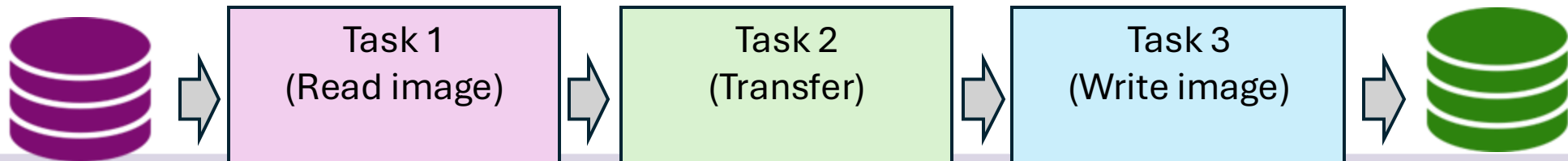
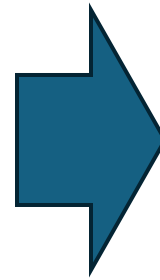


3. Parallel and distributed computing

Discussion: requirements for computing and data processing

Requirements

Example 1: unify different image files

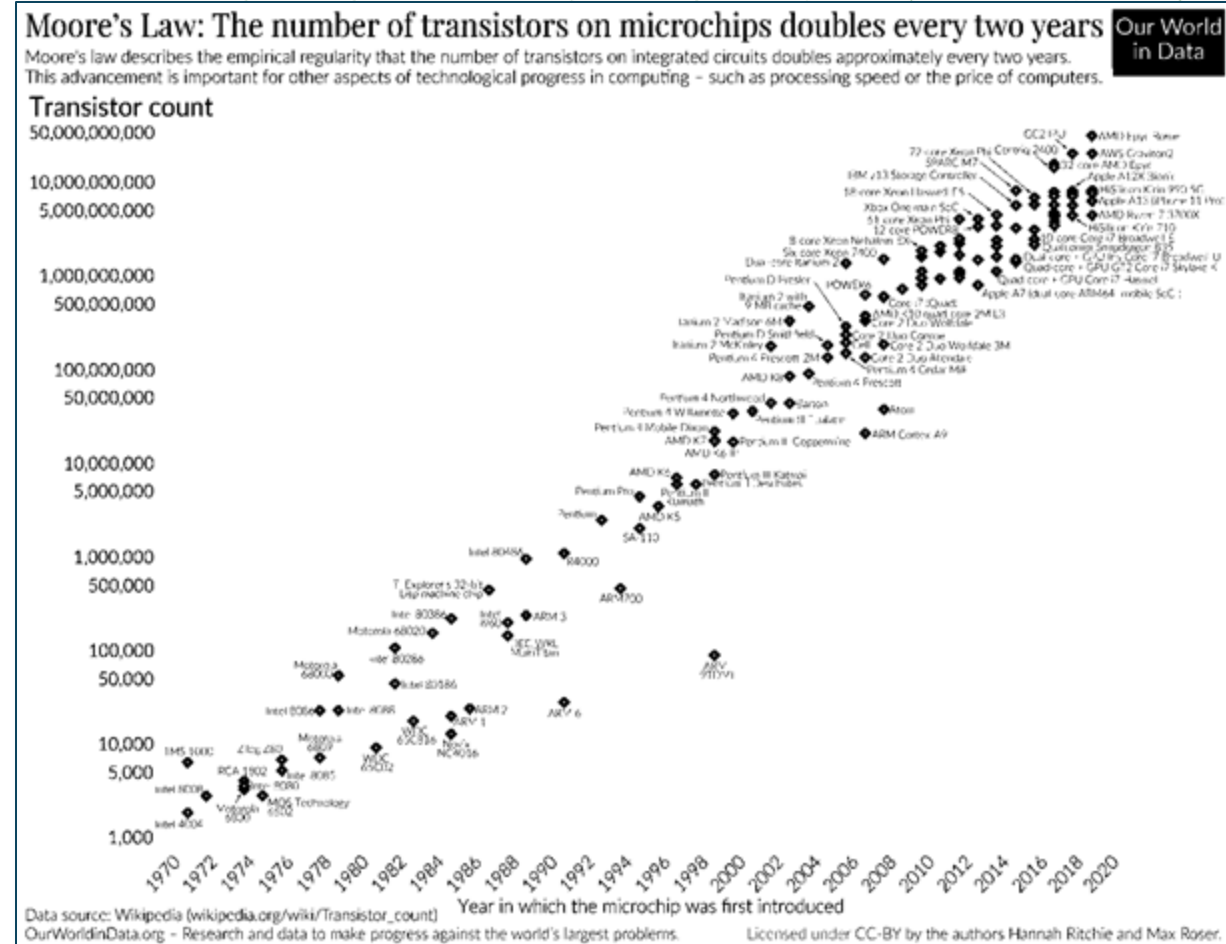
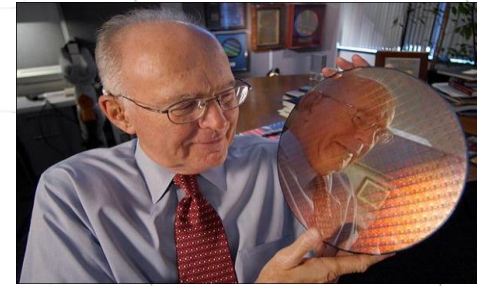


Moore's Law

Gordon Moore (co-founder of Intel) predicted in 1965 that the **transistor density of semiconductor chips would double roughly every 24 months.**

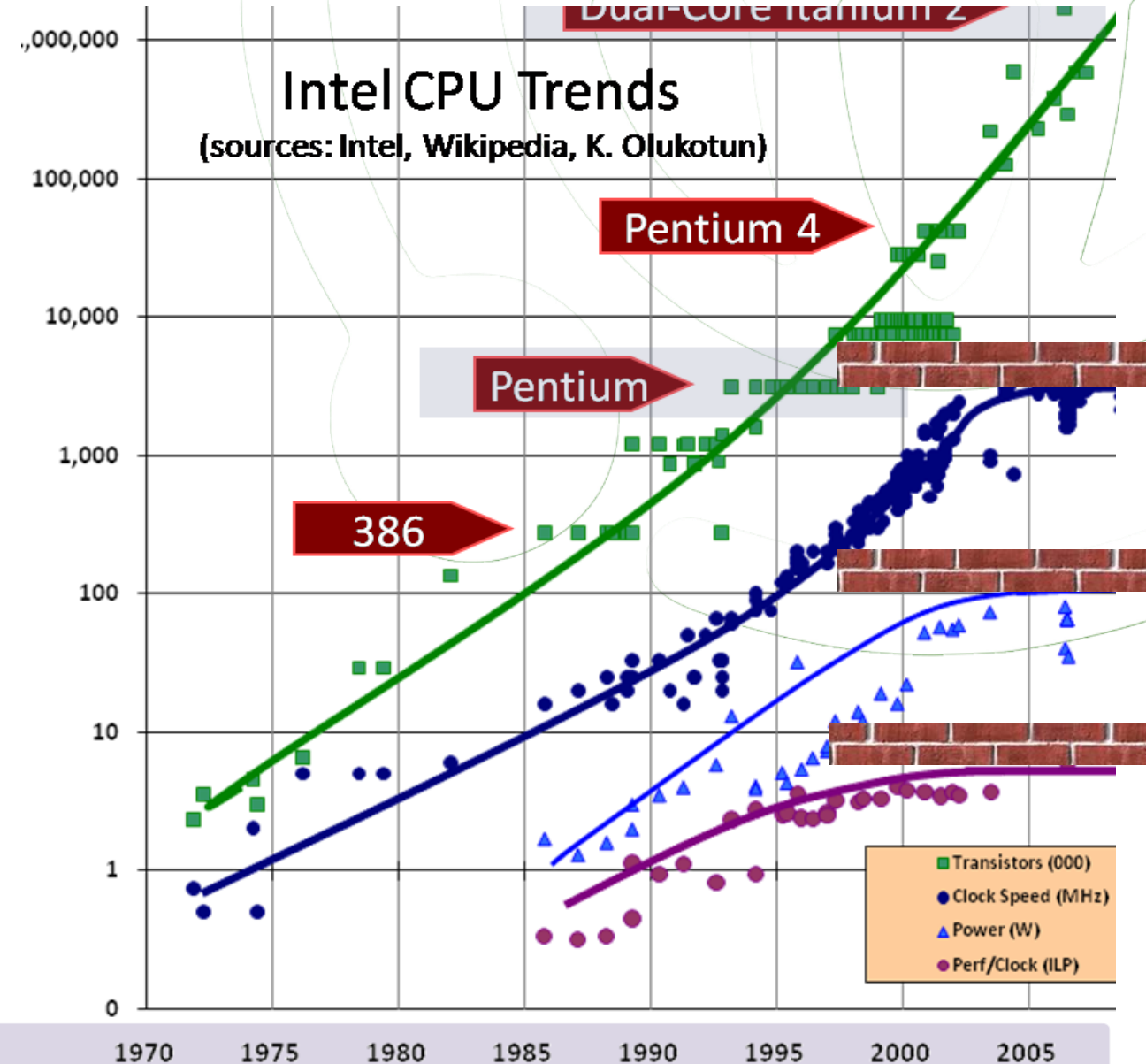
 **More transistors/gates in a chip**

Higher clock frequency

 More advanced **design**

Around 2005: hitting the walls

- 🌐 Power wall
- 🌐 Instruction level parallelism wall
- 🌐 Design complexity wall



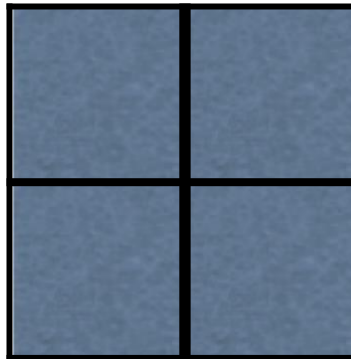
The shift to multi-core



Performance 1
Power 1

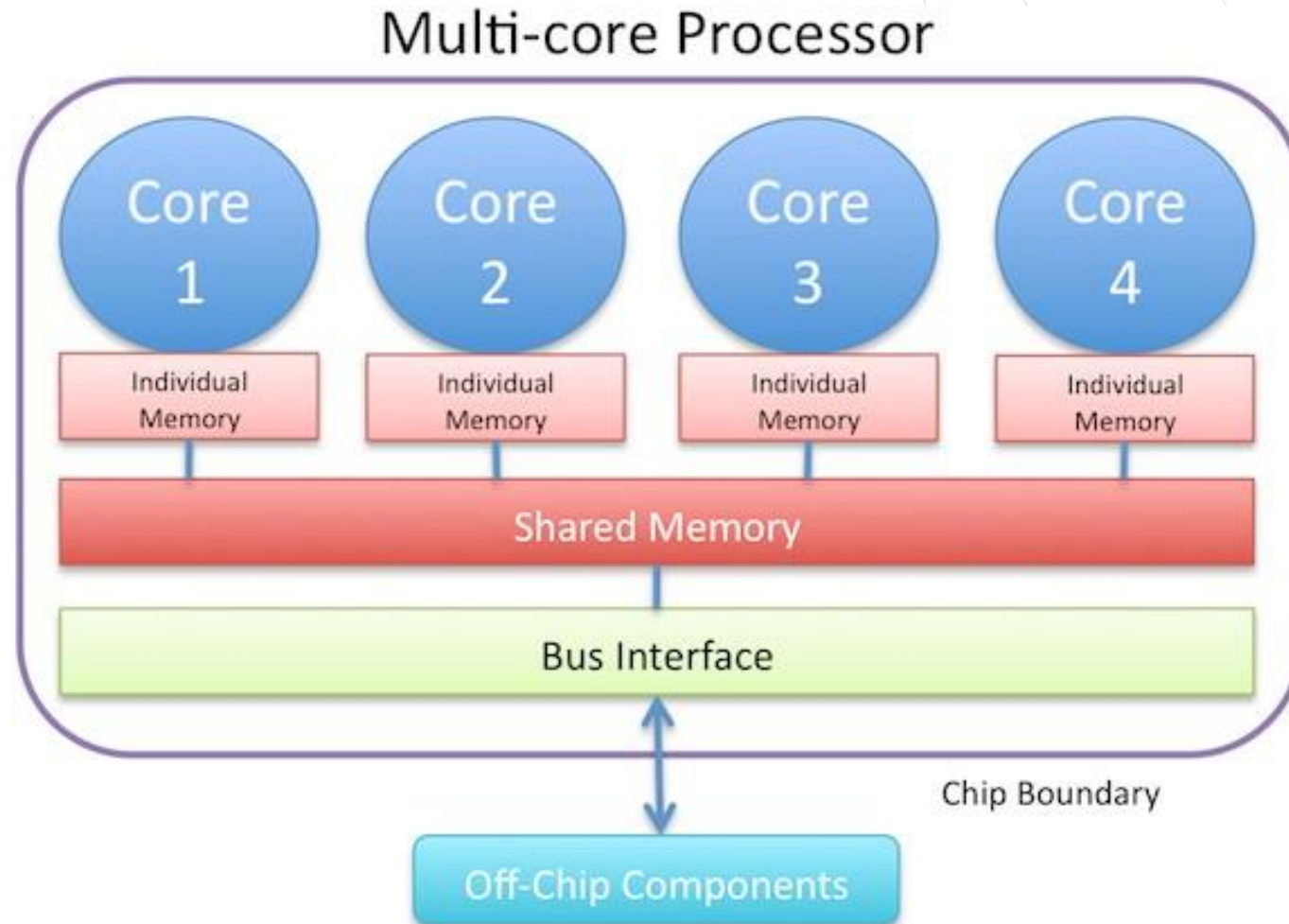


Performance: $\times 2$ ($2 \times F$)
Power: $\times 4$



F remains same
Performance = $\frac{1}{2} \times 4 = 2$
Power = $\frac{1}{4} \times 4 = 1$

Generic multi-core CPU

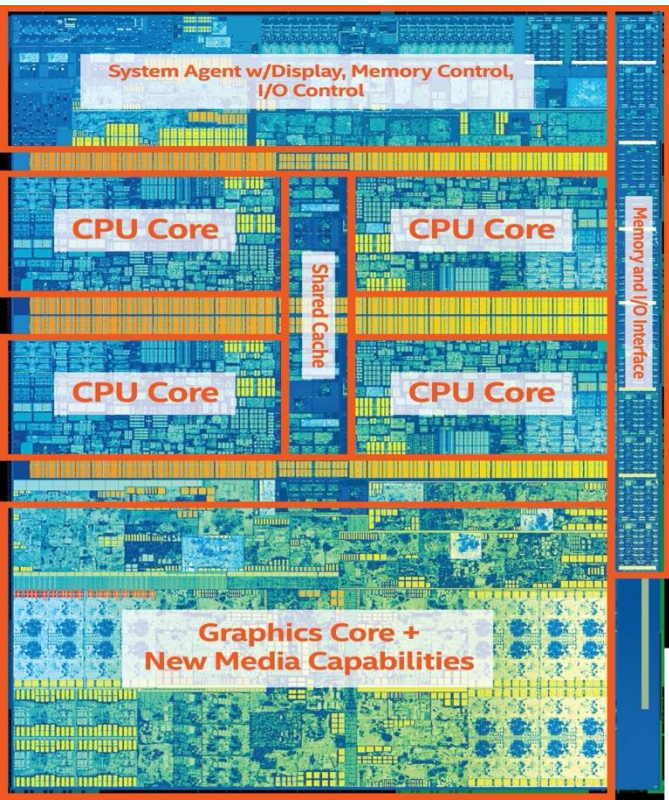


Has been widely used

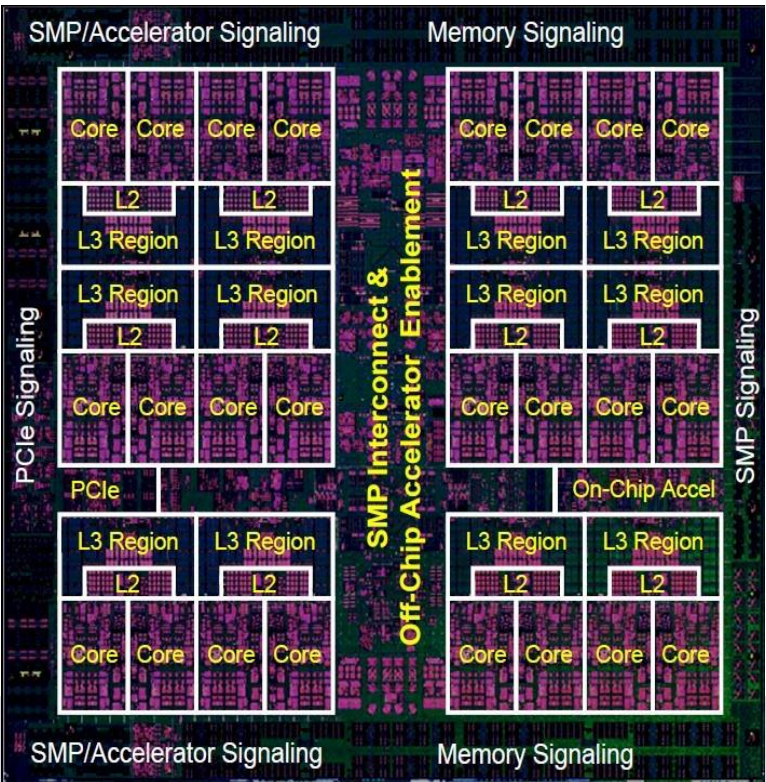


Nvidia Tegra:
Quad-core CPU, 256-core GPU

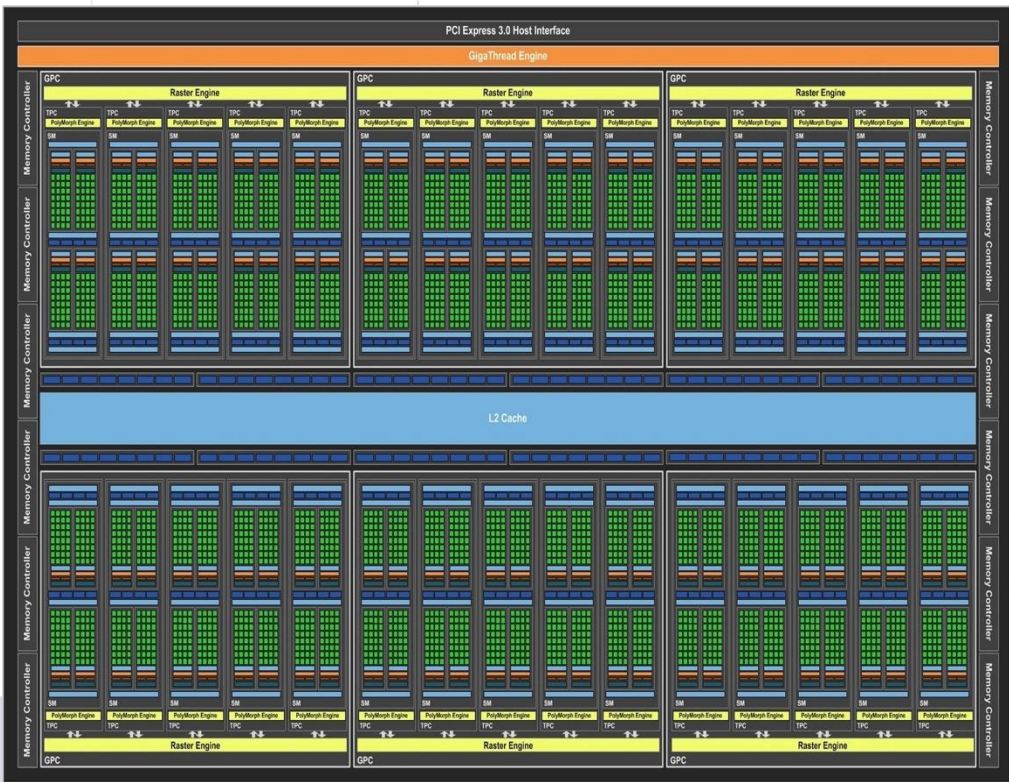
Intel Kaby Lake(2017)



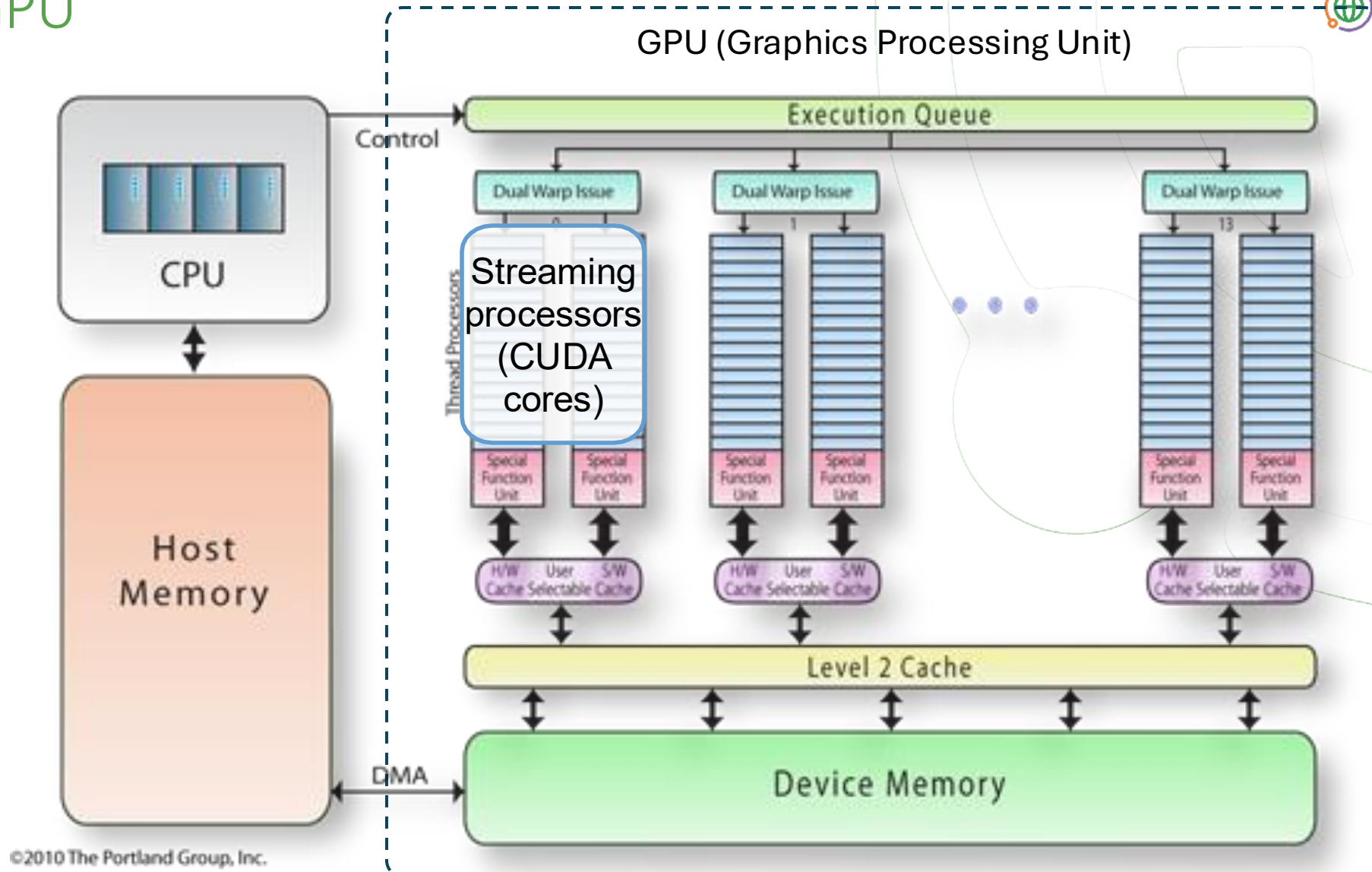
IBM Power9(2016)



Nvidia Titan Xp(2017)



Generic GPU



Streaming processor (SP)
CUDA: computer unified device architecture

Discussion: why parallelism?

Single core performance scaling is over:

- To get better performance than what frequency scaling would provide
- ... Yet, just by waiting until next year, the code would run faster on the next generation of CPUs

We have to adapt to multi/many-core systems:

- Because it is the only way to achieve significantly higher application performance for the foreseeable future

The fastest machine 2023

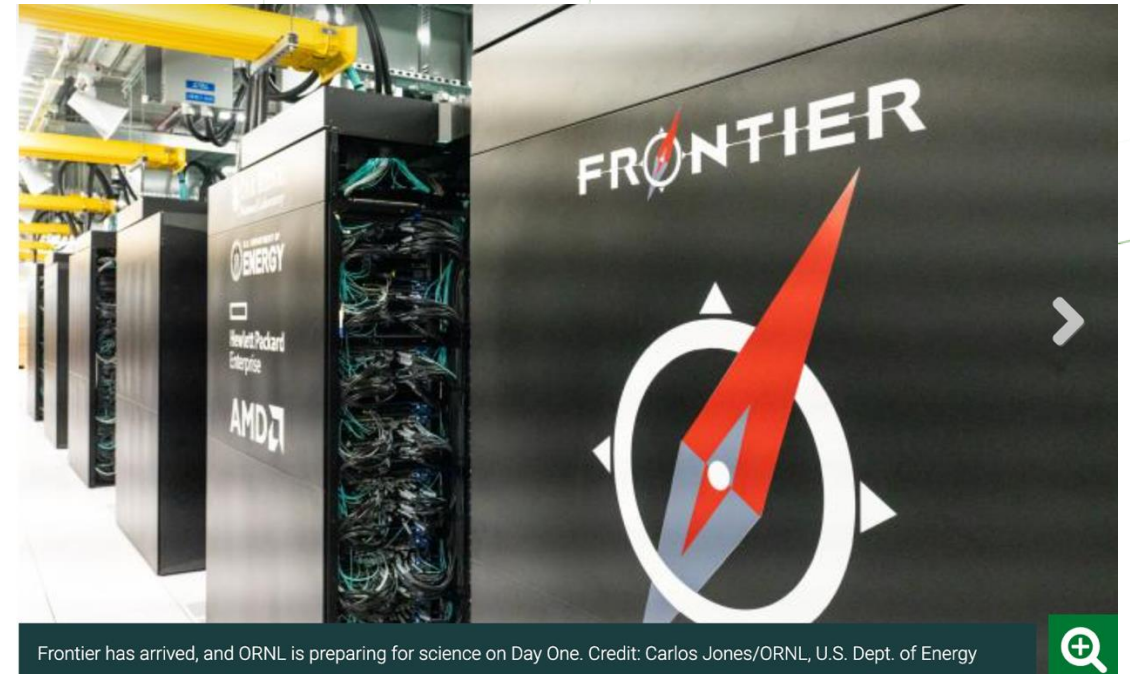
🌐 Vender: HPE Cray

🌐 Core: 8,699,904 (8.7M)

🌐 Peak Flops: 1714 PFlop/s

🌐 Operational Flops: 1206 PFlop./s

🌐 Power: 22,786.00 kW (22MW)



Frontier has arrived, and ORNL is preparing for science on Day One. Credit: Carlos Jones/ORNL, U.S. Dept. of Energy

Data parallization

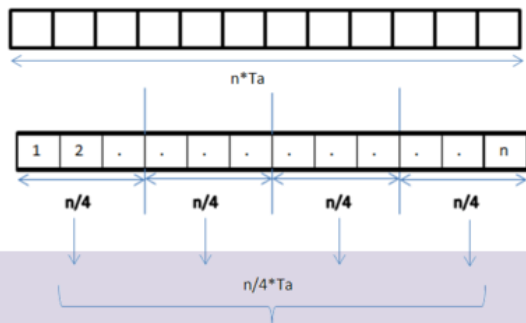
Often used when

- 🌐 The application has to handle **a large data input**
- 🌐 Each of the data points needs to be processed **in the same way** (using the same program)

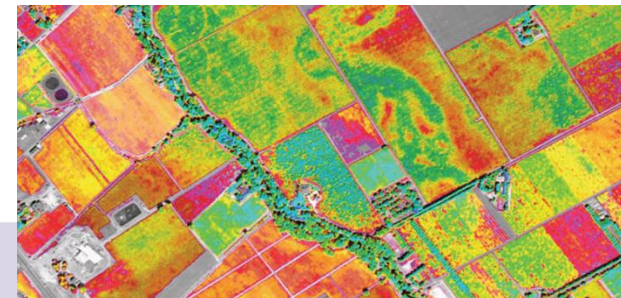
The basic idea

- 🌐 Partition the data into **multiple chunks**
- 🌐 Those data chunks will be processed **in parallel**
- 🌐 Also called **Single Program Multi Data**

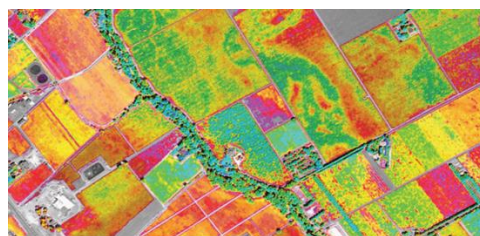
Examples



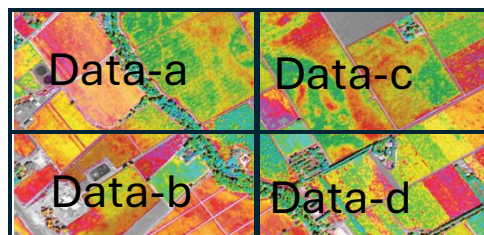
	0	1	2	n-1
0	$a[0][0]$	$a[0][1]$	$a[0][2]$	$a[0][n-1]$
1	$a[1][0]$	$a[1][1]$	$a[1][2]$	$a[1][n-1]$
2	$a[2][0]$	$a[2][1]$	$a[2][2]$	$a[2][n-1]$
3	$a[3][0]$	$a[3][1]$	$a[3][2]$	$a[3][n-1]$
4	$a[4][0]$	$a[4][1]$	$a[4][2]$	$a[4][n-1]$
...
n-1	$a[n-1][0]$	$a[n-1][1]$	$a[n-1][2]$	$a[n-1][n-1]$



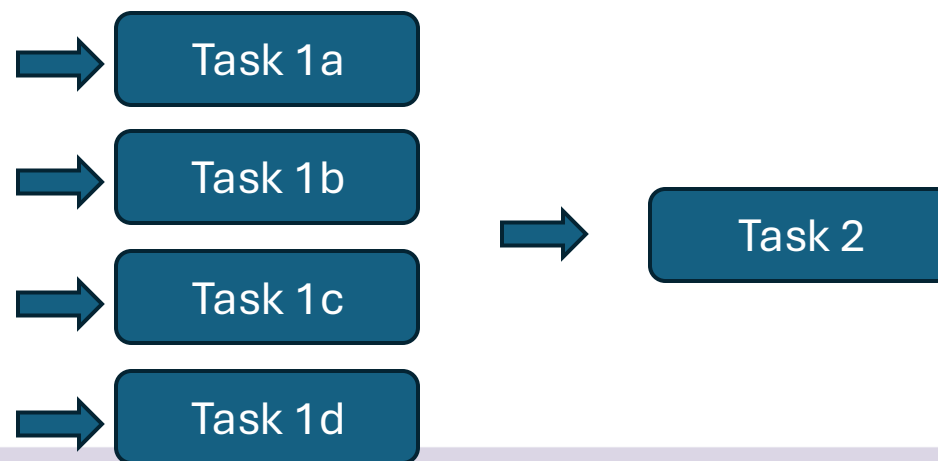
Data parallization



Data



Data



Discussion

 When should we use data parallelization?

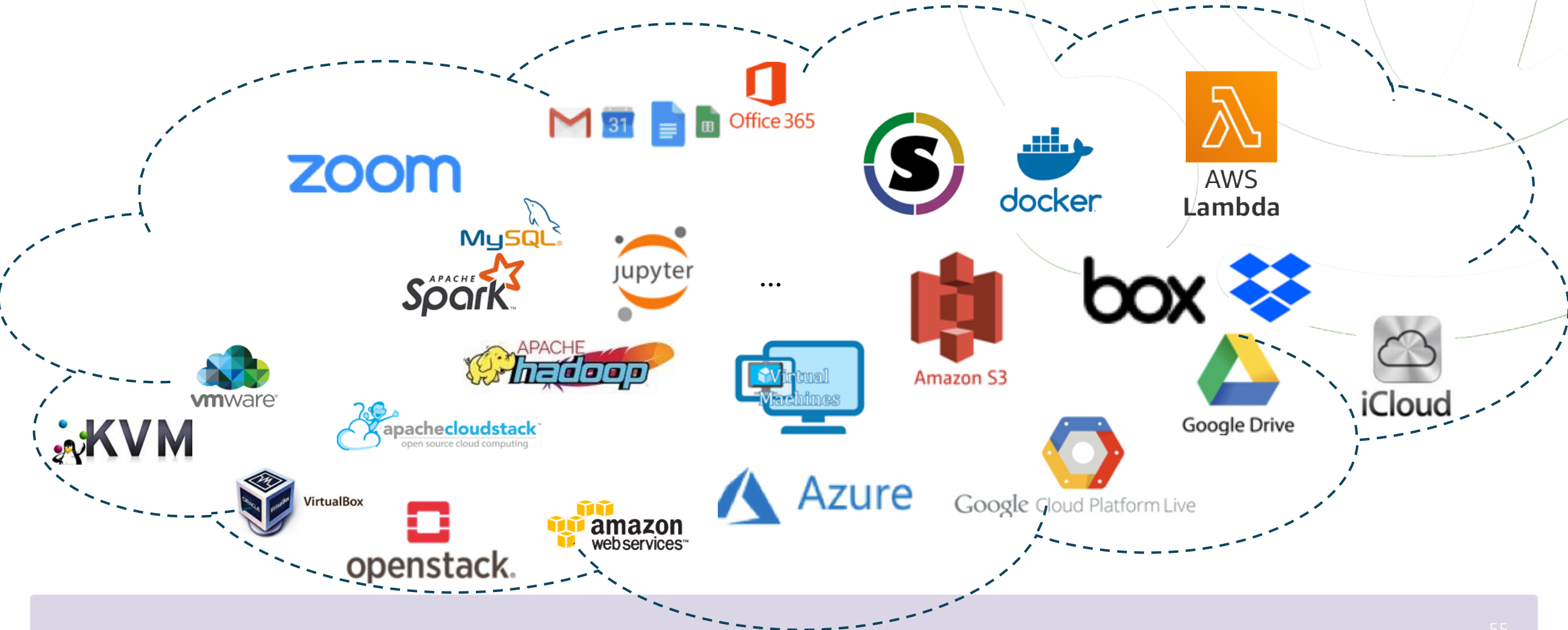
4. Cloud computing and data processing

Discussion: what is cloud computing?

🌐 What is cloud computing?

🌐 Why do we need cloud computing?

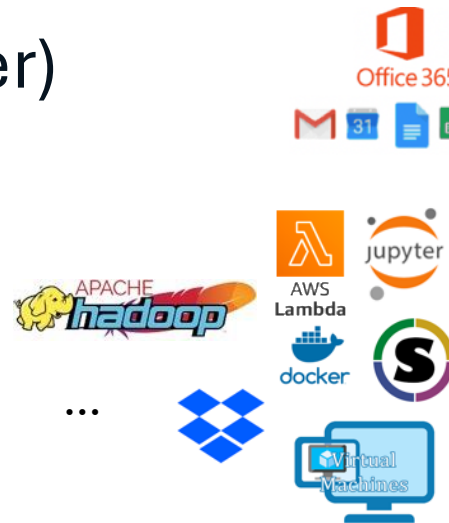
Cloud: services, technologies



What is Cloud?

Services of resources

- Computing power
- Storage
- Network
- Data base
- Server (e.g. Web server)
- Word/Excel/PPT etc.
- ...



Software as a service (SaaS)

Applications, Office, Terminal emulator, etc.

Platform as a service (PaaS)

Data base, web servers, development tools etc.

Infrastructure as a service (IaaS)

Computing elements, storage, network and server etc.

What is Cloud?

 Services of resources

 Access via Internet

- Using internet
- Access from anywhere
- Web portal

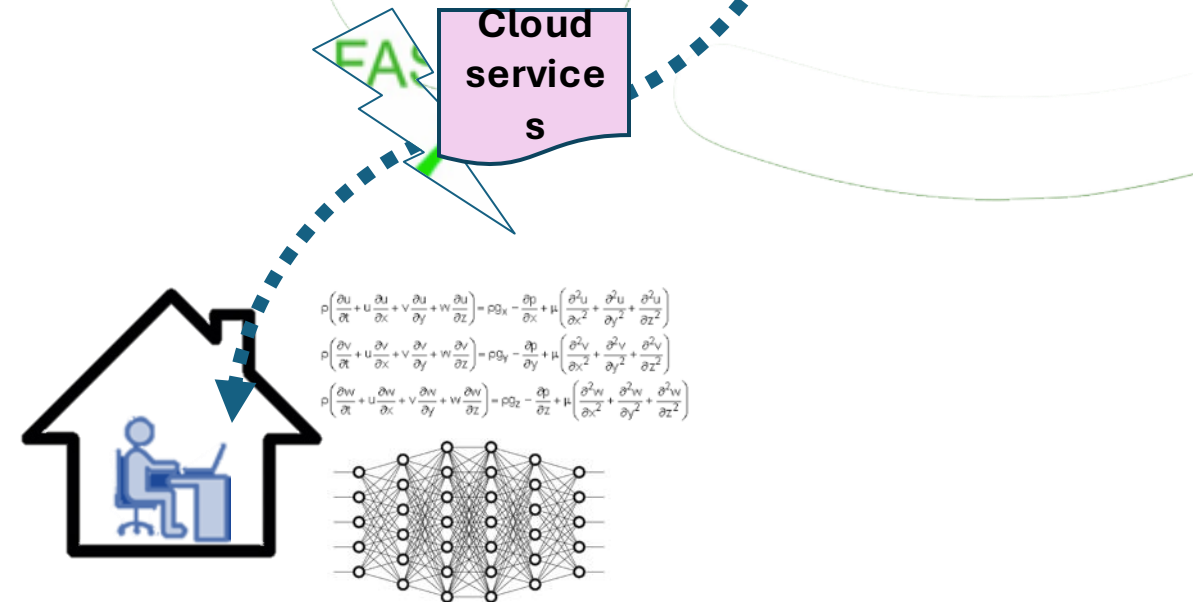


What is Cloud?

Services of resources

Access via Internet

On demand provisioning



What is Cloud?

Services of resources

Access via Internet

On demand provisioning

Flexible price model

- Pay per use
- Pay as you go
- Advanced reserved
- Subscription based



Software as a service (SaaS)
Applications, Office, Terminal emulator, etc.

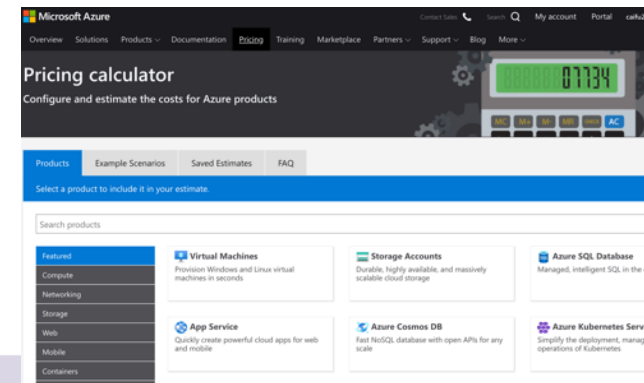
Platform as a service (PaaS)
Data base, web servers, development tools etc.

Infrastructure as a service (IaaS)
Operating elements, storage, network, etc.



Cloud services

FAAS



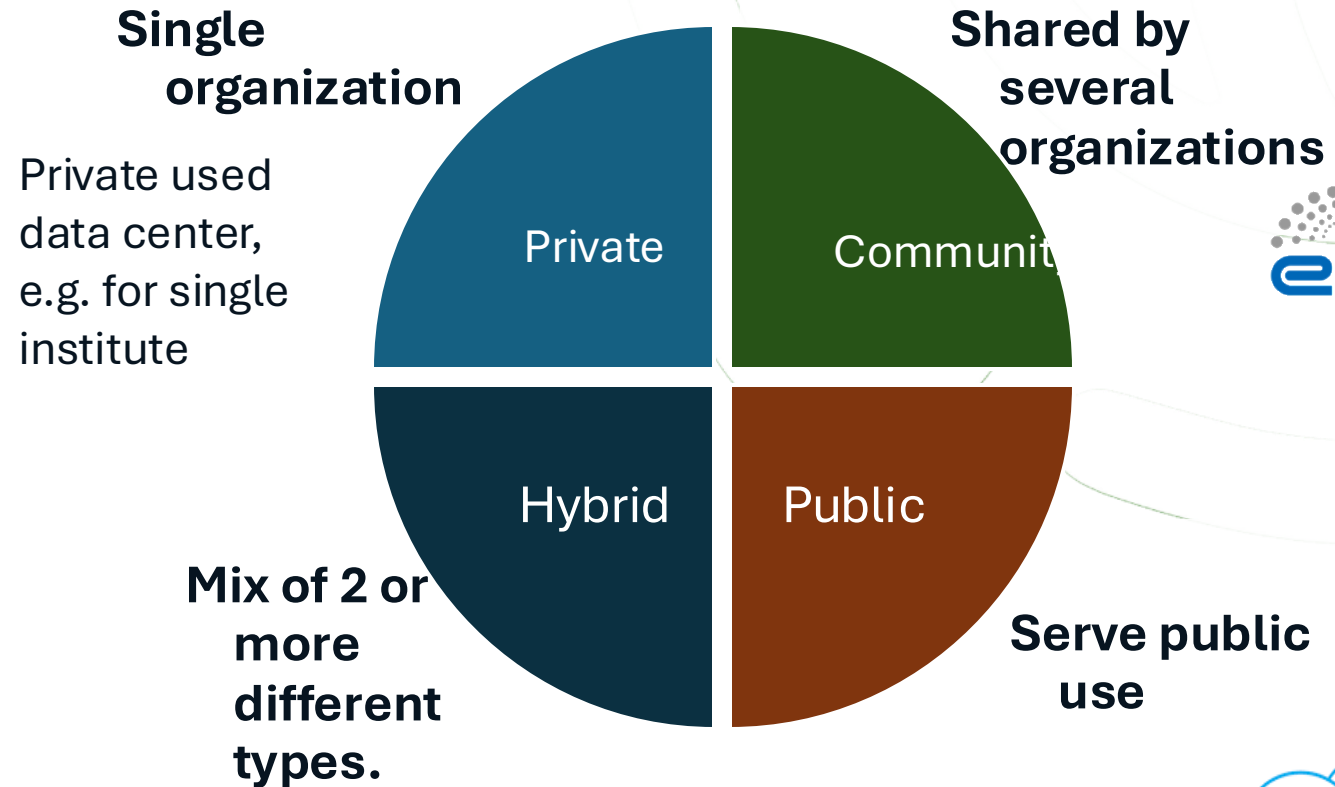
Cloud types

 Private cloud

 Public cloud

 Hybrid cloud

 Community cloud



See more examples from RIs



Cloud and e-Infrastructure

High performance computers

- Super computers
- Clusters

Clouds

- Infrastructure as a service (Virtual Machines)
- Containers
- Platform as a service
- Software as a service

Storage

- Cloud storage, distributed file systems,

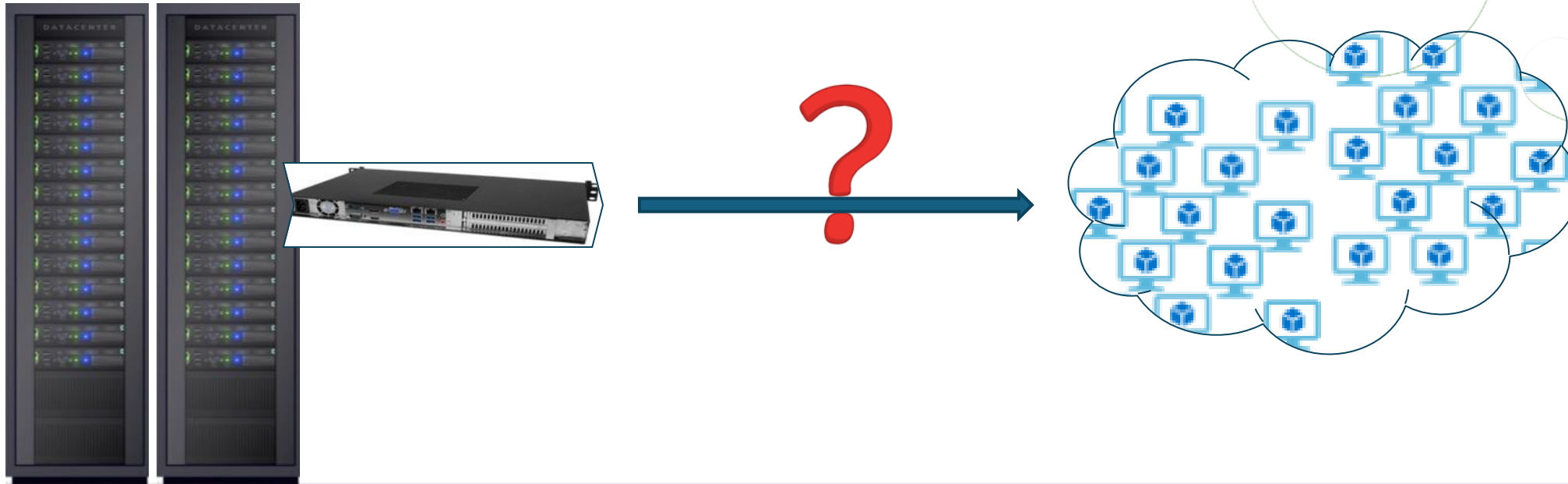
Advanced network

- Light paths
- Software defined networking



How does cloud work?

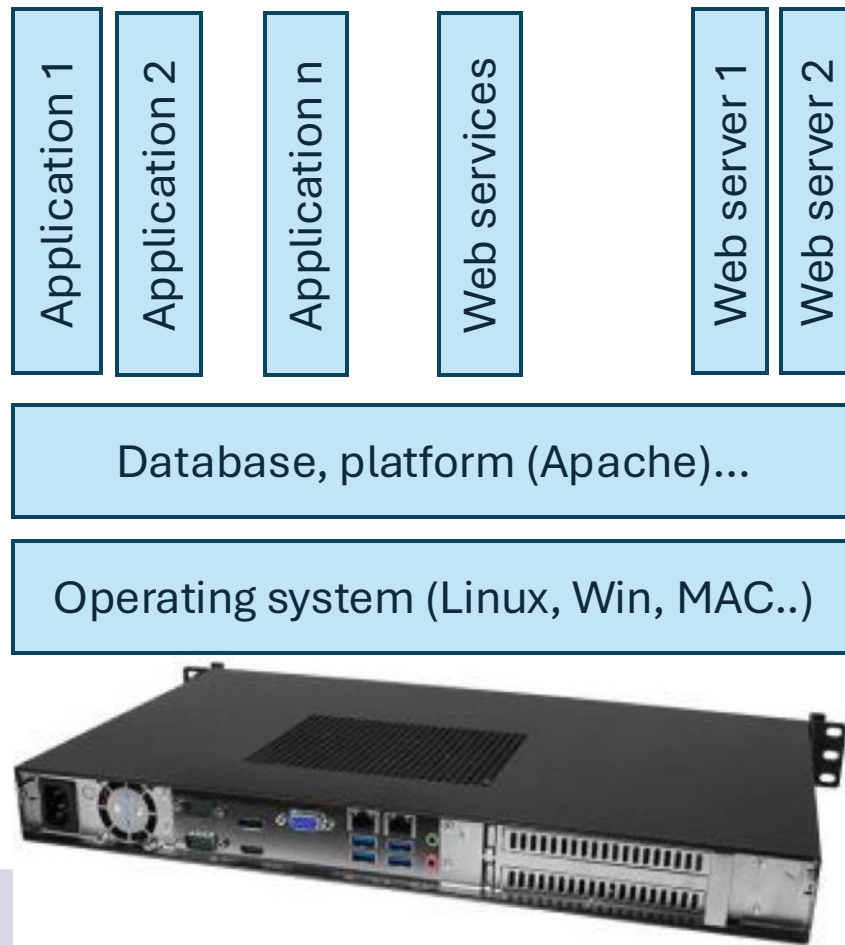
- 🌐 How can I make a big physical machine as many different virtual machines?
- 🌐 How do I handle the requests from different users?



Virtualization

Technology 1: virtualization

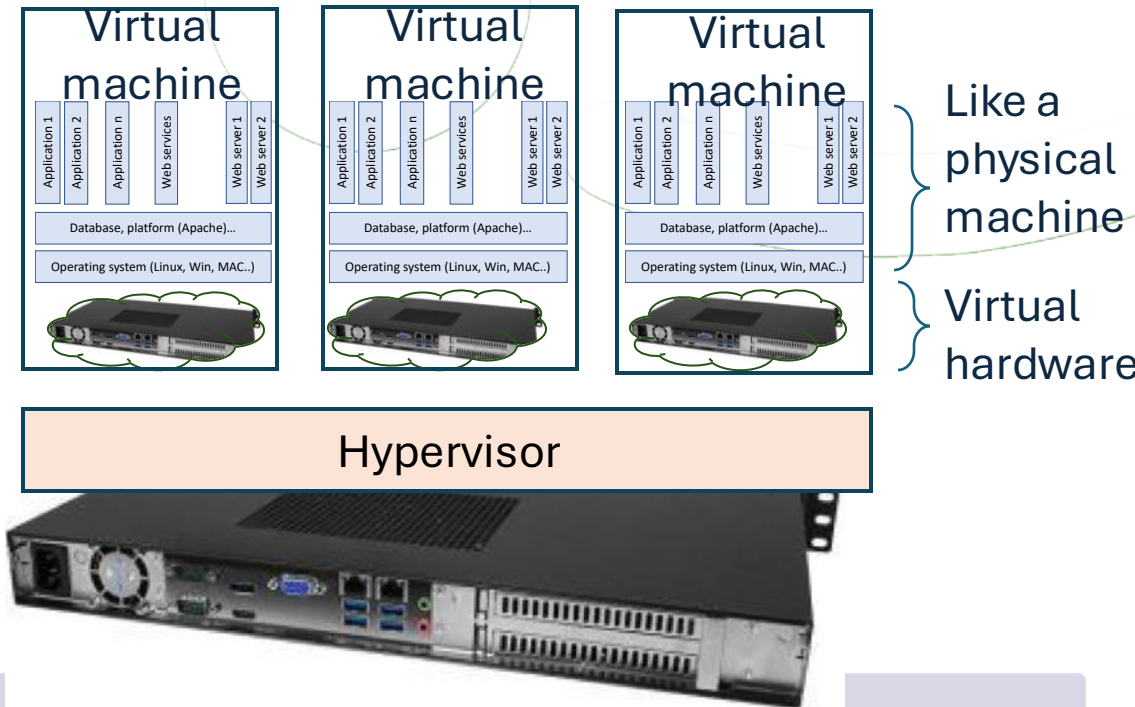
(Virtual Machine and Hypervisor)



A virtual machine (VM):

- A software can emulate the behavior of a real computer
- Contains hardware abstraction, OS kernel, library, file systems and etc.. The file representation is called **VM image**.

Virtualization



Cont.

Technology 1: virtualization

(Virtual Machine and Hypervisor)

A virtual machine (VM):

- A software can emulate the behavior of a real computer
- Contains hardware abstraction, OS kernel, library, file systems and etc. The file representation is called VM

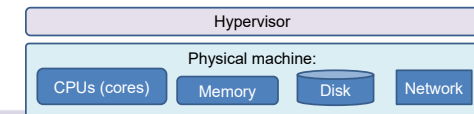
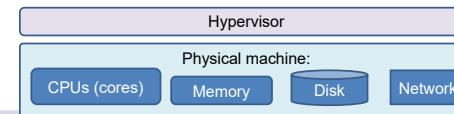
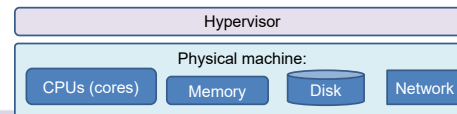
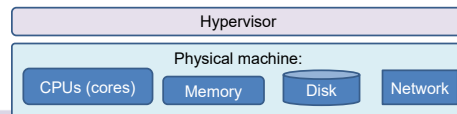
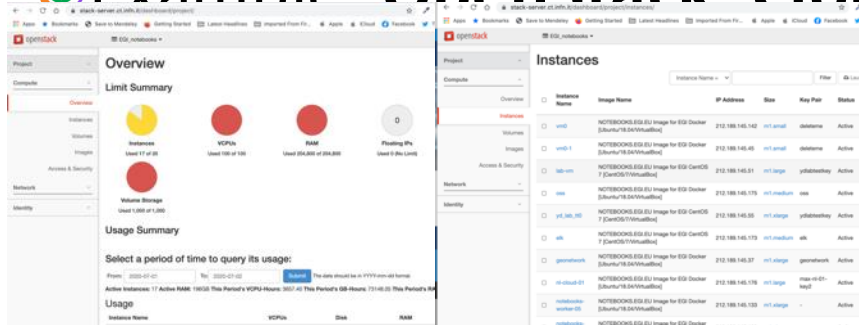
Note:

- ✓ Virtual machines are isolated (VM with different guest OS can run on the same host);
- ✓ **VM images** are files, which are usually in the size of Giga Bytes. Depends on the files included;
- ✓ **VMs** are runtime instances of the VM images in the system;
- ✓ VM images and VMs are dependent on the type of **hypervisor**;
- ✓ One physical machine usually has **only one hypervisor**.



Orchestration

- Allocate the physical resources for different VM requests
 - Provide user interface and API for automating the provisioning of VM requests
 - Allow administrators/users to check the current status of the resources, and manipulate them
- Example: OpenStack, CloudStack, vRealize, Puppet, cloudformation (AWS)**

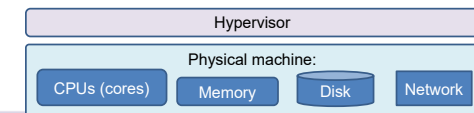
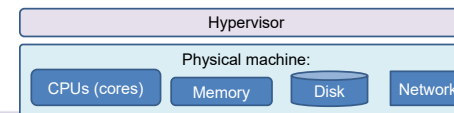
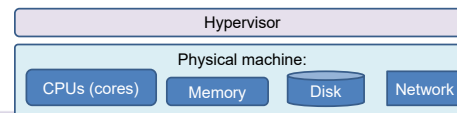
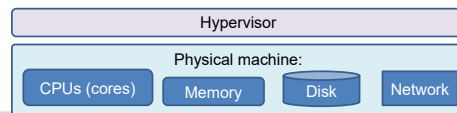


Cont.

- Allocate the physical resources for different VM requests
- Provide user interface and API for automating the provisioning of VM requests

Note:

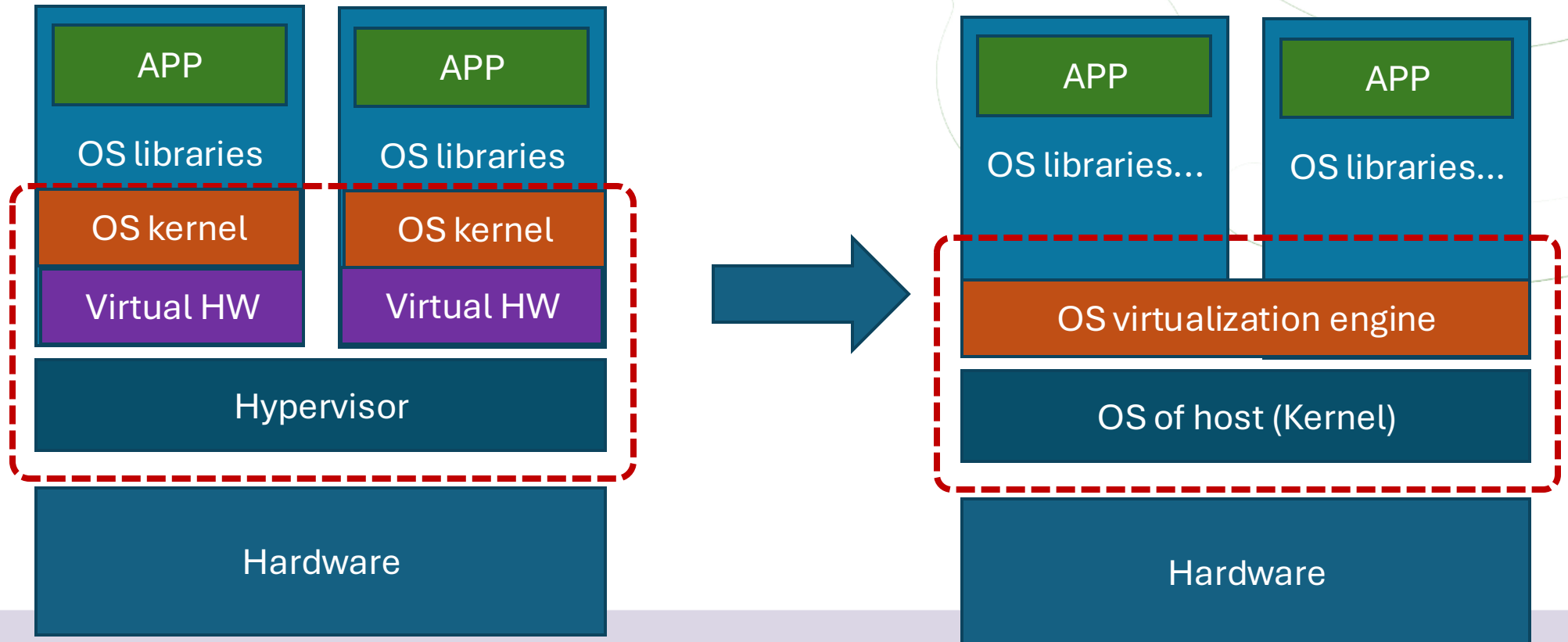
- ✓ An Orchestration system is interacting with the hypervisor, but independent from the hypervisor
- ✓ An Orchestration system provides interface for both users and administrators



Operating system level virtualization

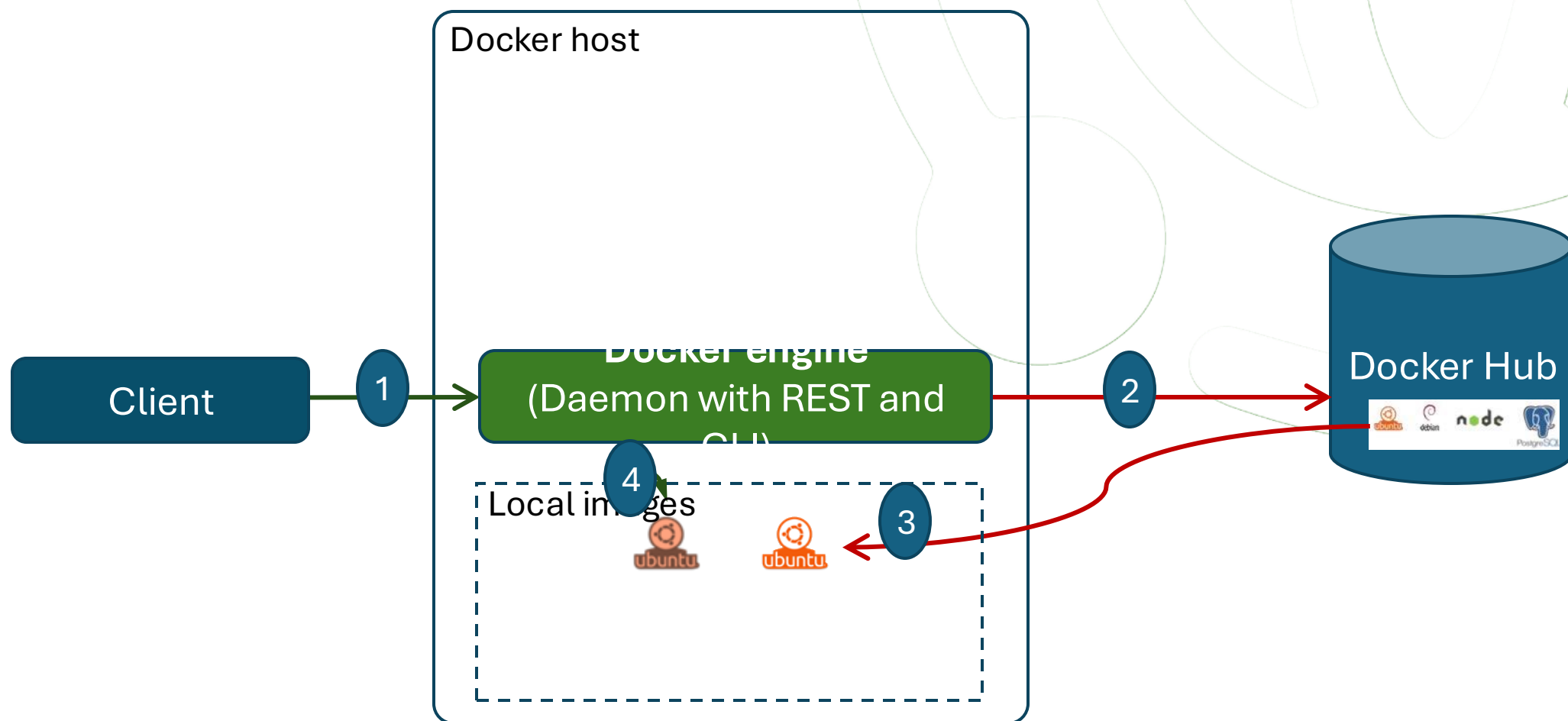
🌐 Reduce full virtual hardware Shared kernel

🌐 From full virtualization to container



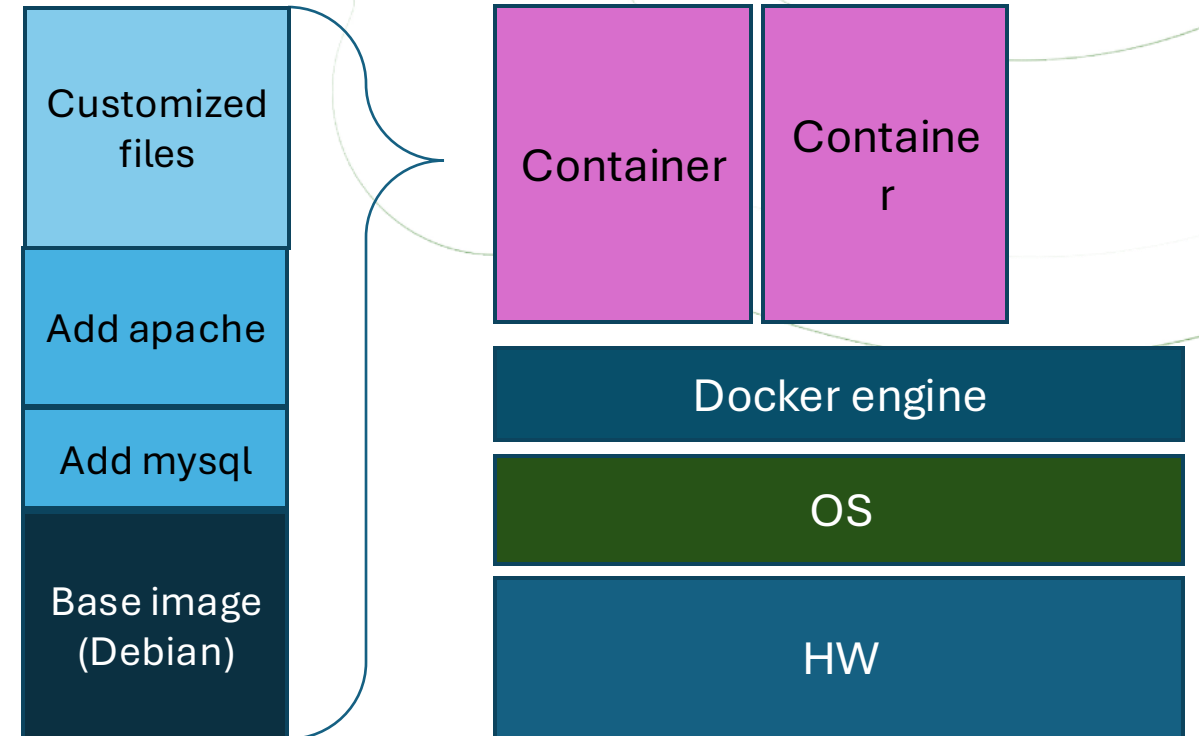
Docker: from image to container

PULL, BUILD



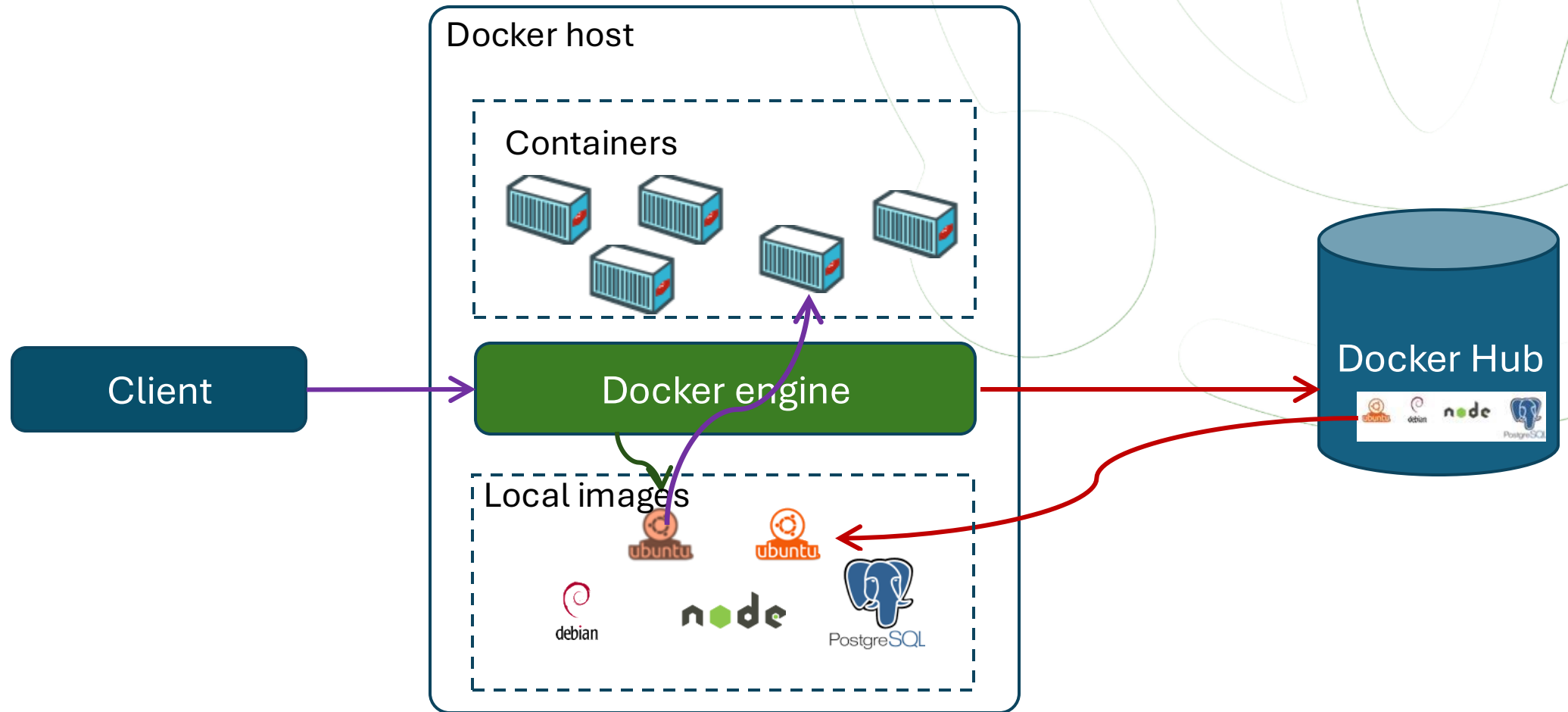
Docker image

- 🌐 Images are comprised of multiple layers,
- 🌐 Every image contains a base layer
- 🌐 Each layer references or is based on another image
- 🌐 Each image contains software you want to run
- 🌐 Basic layers are read only



Docker: from image to container

Multiple containers

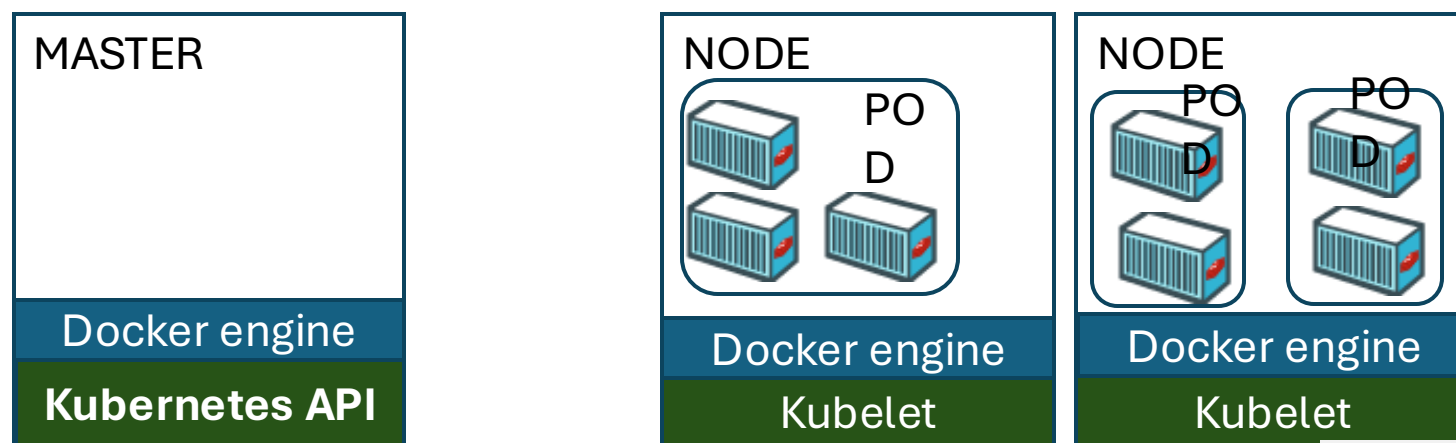


Kubernetes cluster, POD

🌐 Smallest unit in Kubernetes, A structural abstraction of a group Containers

- Some Containers are dependent, and need to be deployed in a single host, or work together. Share IP address or port space.
- Can also be on container per POD

🌐 Containers in a POD share storage/network



<https://kubernetes.io/docs/setup/pick-right-solution>

Kubernetes utilization

- 🌐 Google Kubernetes Engine (GKE)
- 🌐 Amazon Elastic Container Service for Kubernetes
- 🌐 Azure Kubernetes Service (AKS)



Google Container Engine
(GKE)
Google Container Registry



Amazon EKS



Azure Kubernetes
Service (AKS)

What do we have to consider when choosing cloud service? 

 Which provider?

 Which data center?

 What Cloud services?

 What capacity?

 What budget?

 ...

Choose cloud service

The Cloud Portal (e.g. Azure)

An account

Service catalogue

Order the services
and ask for on
demand provision

The screenshot displays the Azure portal interface. At the top, there's a navigation bar with the URL 'portal.azure.com/#home' and a search bar. Below the navigation bar, a sidebar on the left lists various categories: 'Create a resource', 'Home', 'Dashboard', 'All services', 'FAVORITES', 'All resources', 'Resource groups', 'App Services', 'SQL databases', 'Azure Cosmos DB', 'Virtual machines', 'Load balancers', 'Storage accounts', 'Virtual networks', 'Azure Active Directory', 'Monitor', 'Advisor', 'Security Center', 'Help + support', and 'Cost Management + Billing'. The main area is titled 'Welcome to Azure!' and features a grid of service tiles. These tiles are organized into categories like 'COMPUTE', 'NETWORKING', 'STORAGE', 'WEB', 'MOBILE', 'CONTAINERS', 'DATABASES', 'ANALYTICS', 'BLOCKCHAIN', 'INTERNET OF THINGS', 'HYBRID REALITY', 'INTEGRATION', 'IDENTITY', and 'SECURITY'. Each tile contains an icon, a name, and a brief description. At the bottom, there's a row of large icons for 'Virtual machines', 'App Services', 'Storage accounts', 'SQL databases', 'Azure Database for PostgreSQL', and 'Azure Cosmos DB'. The top right corner shows a user profile for 'drzmzhao@gmail.com' with a 'Sign out' button.

Select cloud services

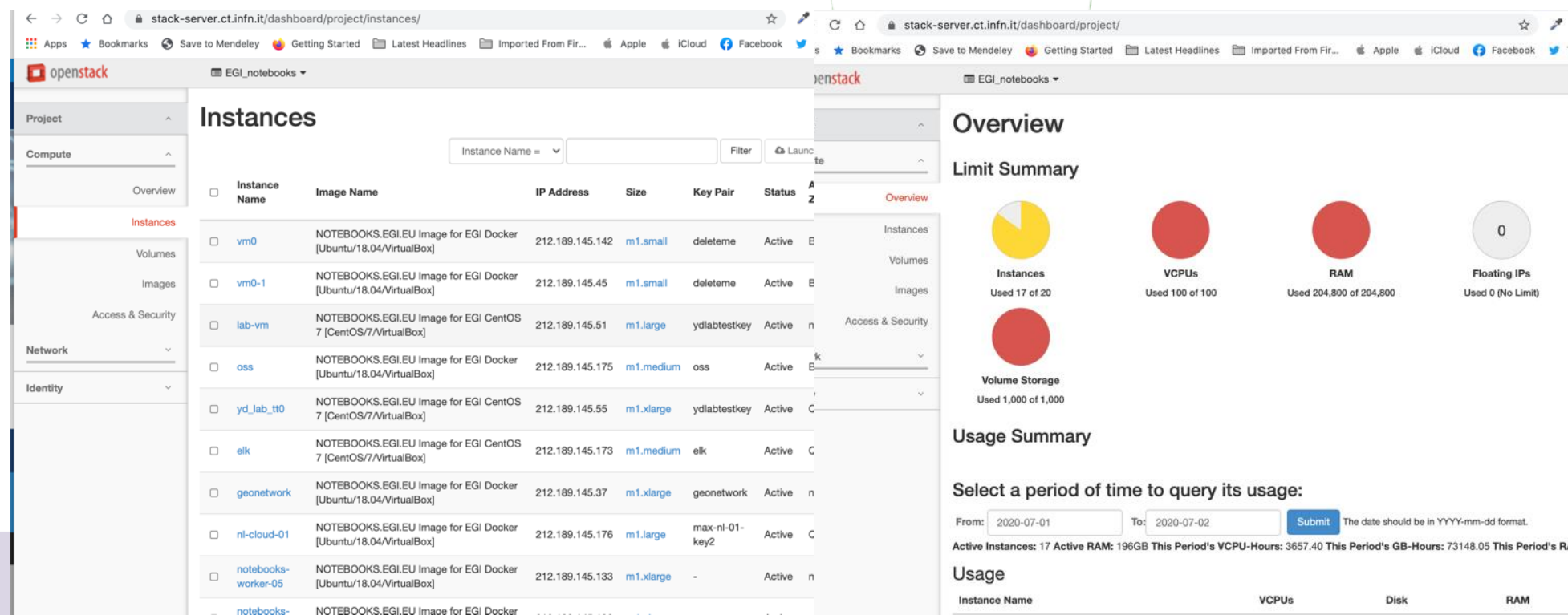


 Dashboard of the available resources

 Instances

 vCPU

 RAM



The screenshot displays the OpenStack dashboard interface. The left sidebar shows navigation options: Project, Compute, Overview, Instances (highlighted in red), Volumes, Images, Access & Security, Network, and Identity. The main content area is titled "Instances" and contains a table of active instances. The table has columns for Instance Name, Image Name, IP Address, Size, Key Pair, Status, and a Z icon. The instances listed are vm0, vm0-1, lab-vm, oss, yd_lab_tt0, elk, geonetwork, nl-cloud-01, notebooks-worker-05, and notebooks-.

Instance Name	Image Name	IP Address	Size	Key Pair	Status	Z
vm0	NOTEBOOKS.EGI.EU Image for EGI Docker [Ubuntu/18.04/VirtualBox]	212.189.145.142	m1.small	delete	Active	B
vm0-1	NOTEBOOKS.EGI.EU Image for EGI Docker [Ubuntu/18.04/VirtualBox]	212.189.145.45	m1.small	delete	Active	B
lab-vm	NOTEBOOKS.EGI.EU Image for EGI CentOS 7 [CentOS/7/VirtualBox]	212.189.145.51	m1.large	yclabtestkey	Active	n
oss	NOTEBOOKS.EGI.EU Image for EGI Docker [Ubuntu/18.04/VirtualBox]	212.189.145.175	m1.medium	oss	Active	B
yd_lab_tt0	NOTEBOOKS.EGI.EU Image for EGI CentOS 7 [CentOS/7/VirtualBox]	212.189.145.55	m1.xlarge	yclabtestkey	Active	C
elk	NOTEBOOKS.EGI.EU Image for EGI CentOS 7 [CentOS/7/VirtualBox]	212.189.145.173	m1.medium	elk	Active	C
geonetwork	NOTEBOOKS.EGI.EU Image for EGI Docker [Ubuntu/18.04/VirtualBox]	212.189.145.37	m1.xlarge	geonetwork	Active	n
nl-cloud-01	NOTEBOOKS.EGI.EU Image for EGI Docker [Ubuntu/18.04/VirtualBox]	212.189.145.176	m1.large	max-nl-01-key2	Active	C
notebooks-worker-05	NOTEBOOKS.EGI.EU Image for EGI Docker [Ubuntu/18.04/VirtualBox]	212.189.145.133	m1.xlarge	-	Active	n
notebooks-	NOTEBOOKS.EGI.EU Image for EGI Docker					

On the right side of the dashboard, there is an "Overview" section with a "Limit Summary" and a "Usage Summary". The "Limit Summary" shows four metrics: Instances (Used 17 of 20), VCPUs (Used 100 of 100), RAM (Used 204,800 of 204,800), and Floating IPs (Used 0 of No Limit). The "Usage Summary" section includes a date range selector (From: 2020-07-01, To: 2020-07-02) and a "Submit" button. Below the date range, it displays "Active Instances: 17", "Active RAM: 196GB", "This Period's VCPU-Hours: 3657.40", "This Period's GB-Hours: 73148.05", and "This Period's RA".

Cont.

A big list of services they offer

Including IaaS, PaaS, SaaS, and other new items

VM example

- Configure type
- Data center
- OS
- Disk
- Network

The screenshot displays the Microsoft Azure portal interface. On the left, a dark sidebar contains a navigation menu with options like 'Create a resource', 'Home', 'Dashboard', 'All services', 'FAVORITES', 'All resources', 'Resource groups', 'Quickstart Center', 'App Services', 'Function App', 'SQL databases', 'Azure Cosmos DB', 'Virtual machines', 'Load balancers', 'Storage accounts', 'Virtual networks', 'Azure Active Directory', 'Monitor', 'Advisor', 'Security Center', and 'Help + support'. The 'Virtual machines' option is highlighted. The main content area shows the 'Create a virtual machine' page. The breadcrumb navigation is 'Home > Virtual machines >'. The page title is 'Create a virtual machine'. Below the title, there is a description: 'Create a virtual machine that runs Linux or Windows. Select an image from Azure marketplace or use your own customized image. Complete the Basics tab then Review + create to provision a virtual machine with default parameters or review each tab for full customization. [Learn more](#)'. The form is divided into sections: 'Project details' with 'Subscription' (set to 'Azure subscription 1') and 'Resource group' (set to '(New) Resource group' with a 'Create new' link); 'Instance details' with 'Virtual machine name' (empty), 'Region' (set to '(US) West US'), 'Availability options' (set to 'No infrastructure redundancy required'), 'Image' (set to 'Ubuntu Server 18.04 LTS' with a 'Browse all public and private images' link), and 'Azure Spot instance' (set to 'No'). At the bottom, there are buttons for 'Review + create', '< Previous', and 'Next : Disks >'. The top of the browser window shows the URL 'portal.azure.com/#create/Microsoft.VirtualMachine' and various browser tabs and extensions.

Discussion




 What is difference between Cloud and a normal computer?

5. Running applications in Cloud





Discussion: How to run your application on cloud?

Options for running scientific application in cloud

Typical options:

-  Infrastructure as a service (e.g., VM, Storage, Network)
-  Platform as a service (e.g., Database, big data cluster)
-  Software as a service (e.g., Jupyter Hub)

Many new services...

-  Serverless (Lambda or Function)
-  Blockchain as service
-  DevOps
- 

Types of scientific application

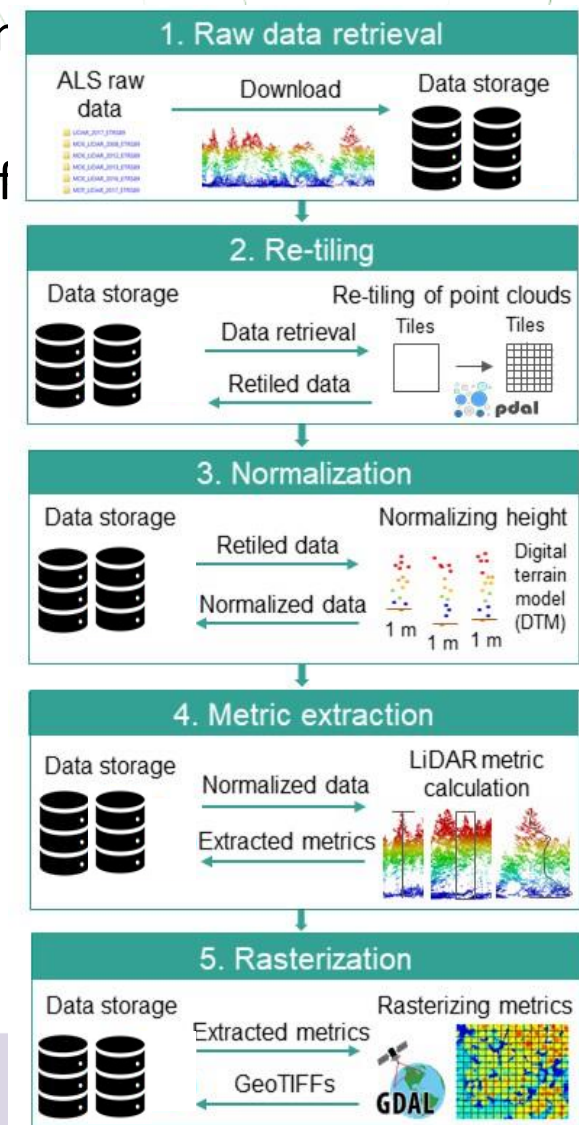
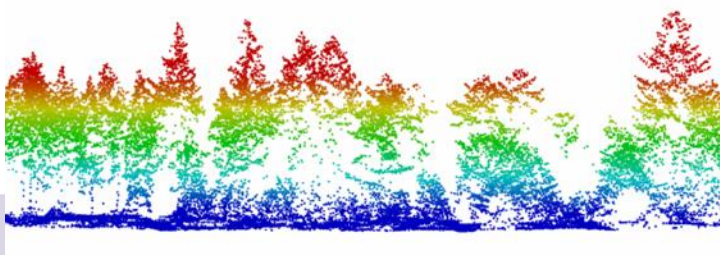
Types of scientific application

- Modelling and simulation,
- big data analytics,
- machine learning,
- sensor data processing,
- data base
- Workflow of different tasks
- ...

An example

- 🌐 A use case from LifeWatch: processing high-resolution Light Detection and Ranging (LiDAR) measurements
- 🌐 A researcher developed basic analysis code in python: point cloud processing of Airborne Laser Scan (ALS) raw data
 - Retiling → normalize height → Lidar metric calculation → Rasterizing
- 🌐 Python code developed in Jupyter, only with data from NL
- 🌐 Executed on our local infrastructure

Processing LiDAR point clouds
(ALS raw data)



Why use cloud?

For larger data set, for instance

- Large Volume data set (e.g., Process ALS data from the entire country or EU scale)
- and Multi data sources (e.g., species information from GBIF)

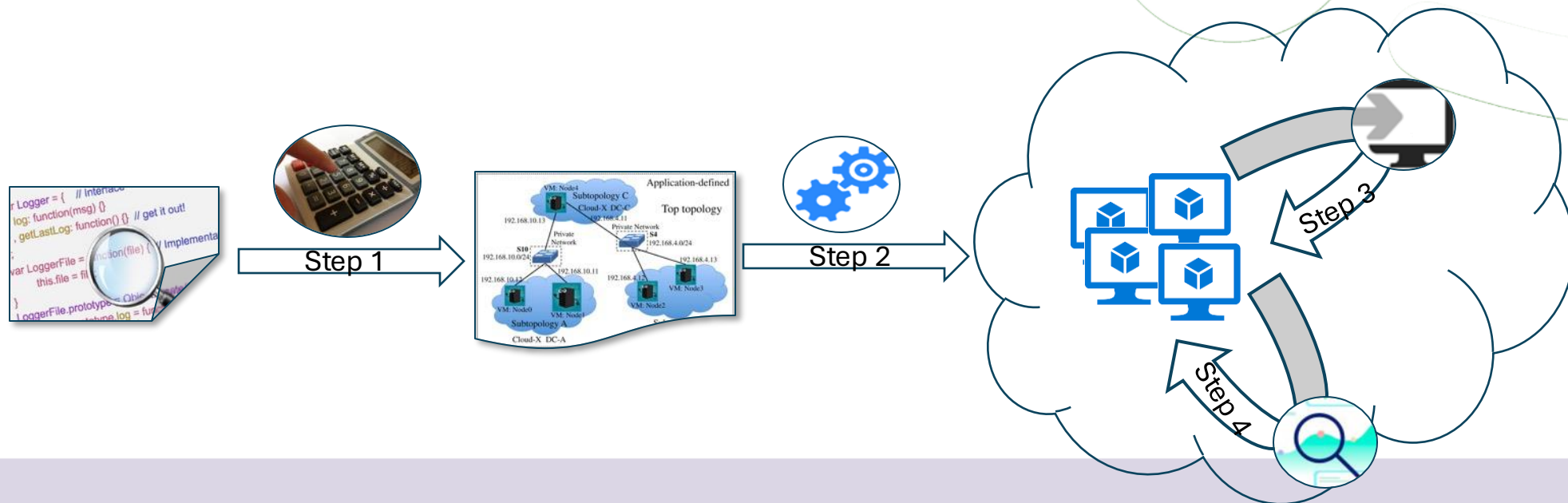
For more intensive computing tasks, for instance

- simulating high resolution models or training high quality deep neural networks (e.g., combining species distribution or climate information)

For combining new features, or models

Option 1: Using Infrastructure services (Virtual machines)

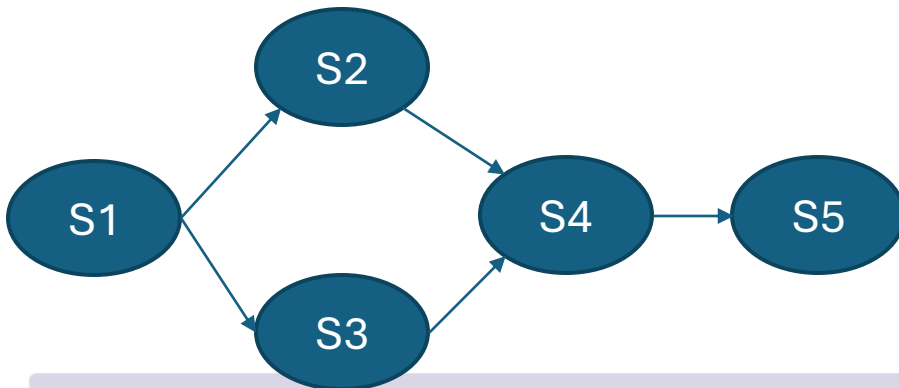
1. Plan Virtual infrastructure (what capacity)
2. Provision Virtual infrastructure (you ask the provider to do it for you)
3. Deploy the Platform and software (you will do it)
4. Execute and monitor the Application (you will do it)



Option 2: if using services

If the tasks are web services (need to be persistantly online)

- Select virtual machines for individual web service, or for group of services based on their performance characteristics (following similar approach for a single application)
- Or set up a container cluster based on a set of virtual machines, and deploy services as containers on the cluster
- Note: if you want to enable auto-scaling, extra capacity needs to reserved



Cont.

- 🌐 The VMs got from Cloud providers are often for general purpose; you can customize the OS version, and hardware configurations
- 🌐 The software platforms needed by your scientific application, e.g., python, java, etc., must be installed by yourself
- 🌐 Do it manually
 - Remotely login the system (as you will try during the lab)
 - Install them using relevant commands on Linux or other systems
- 🌐 Or automated
 - Compose the installation orders as a playbook, and automate it using the tool like ansible

Deploy a distributed application in a remote environment is time consuming!

Using containers

- 🌐 VM images are complete self-contained, but are usually very large;
- 🌐 Application virtualization are not generic for all languages
- 🌐 Container technologies are getting popular in cloud computing.

Virtual machine images

- ✓ Application
- ✓ Platform and libraries
- ✓ File systems
- ✓ Operating systems (full)
- ✓ Hardware configuration and abstraction
- ✓ **Directly deploy above hypervisor**
- ✓ *No other installation needs*

Container (Docker) images

- ✓ Application
- ✓ Platform and libraries
- ✓ File systems
- ✓ Operating systems libraries (without kernel)
- ✓ **Require a container (e.g., docker) engine**
- ✓ **Require operating system kernel from the host environment**
- ✓ *(Virtual) **Hardware** configuration*

Application packages (JAR, WAR)

- ✓ Application
- ✓ Platform and libraries
- ✓ **Require special environment (e.g., java runtime environment)**
- ✓ *Require **full** operating system kernel from the host environment*
- ✓ *(Virtual) **Hardware** configuration*

How does a docker work?

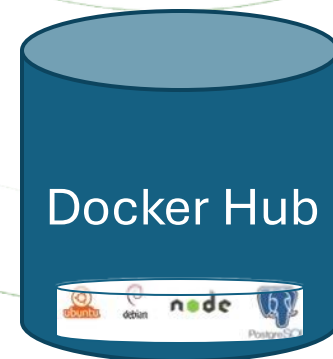
Client

D

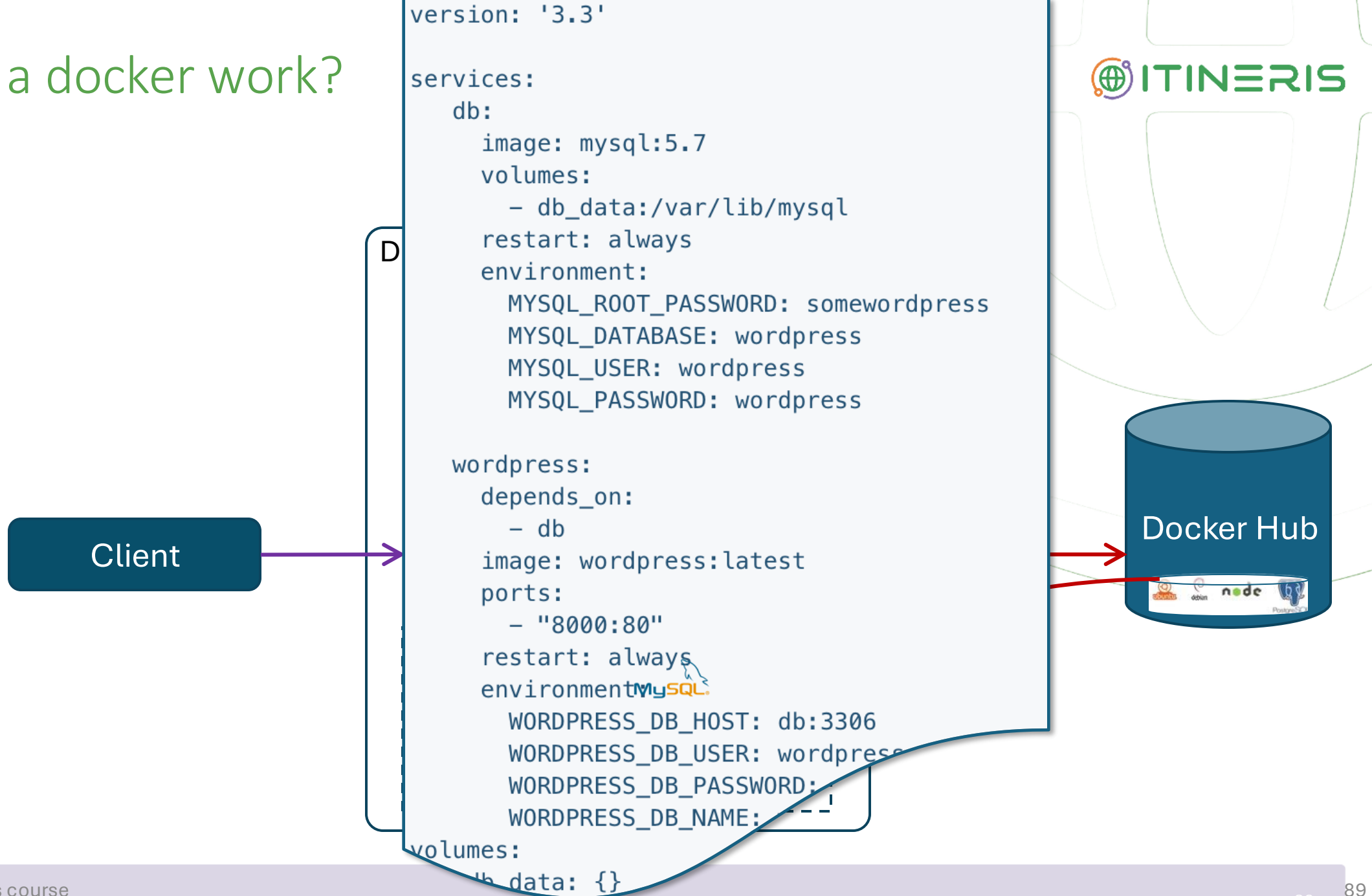
```
version: '3.3'

services:
  db:
    image: mysql:5.7
    volumes:
      - db_data:/var/lib/mysql
    restart: always
    environment:
      MYSQL_ROOT_PASSWORD: somewordpress
      MYSQL_DATABASE: wordpress
      MYSQL_USER: wordpress
      MYSQL_PASSWORD: wordpress

  wordpress:
    depends_on:
      - db
    image: wordpress:latest
    ports:
      - "8000:80"
    restart: always
    environment:
      WORDPRESS_DB_HOST: db:3306
      WORDPRESS_DB_USER: wordpress
      WORDPRESS_DB_PASSWORD: wordpress
      WORDPRESS_DB_NAME: wordpress
    volumes:
      - db_data:/var/lib/mysql
```



How does a docker work?

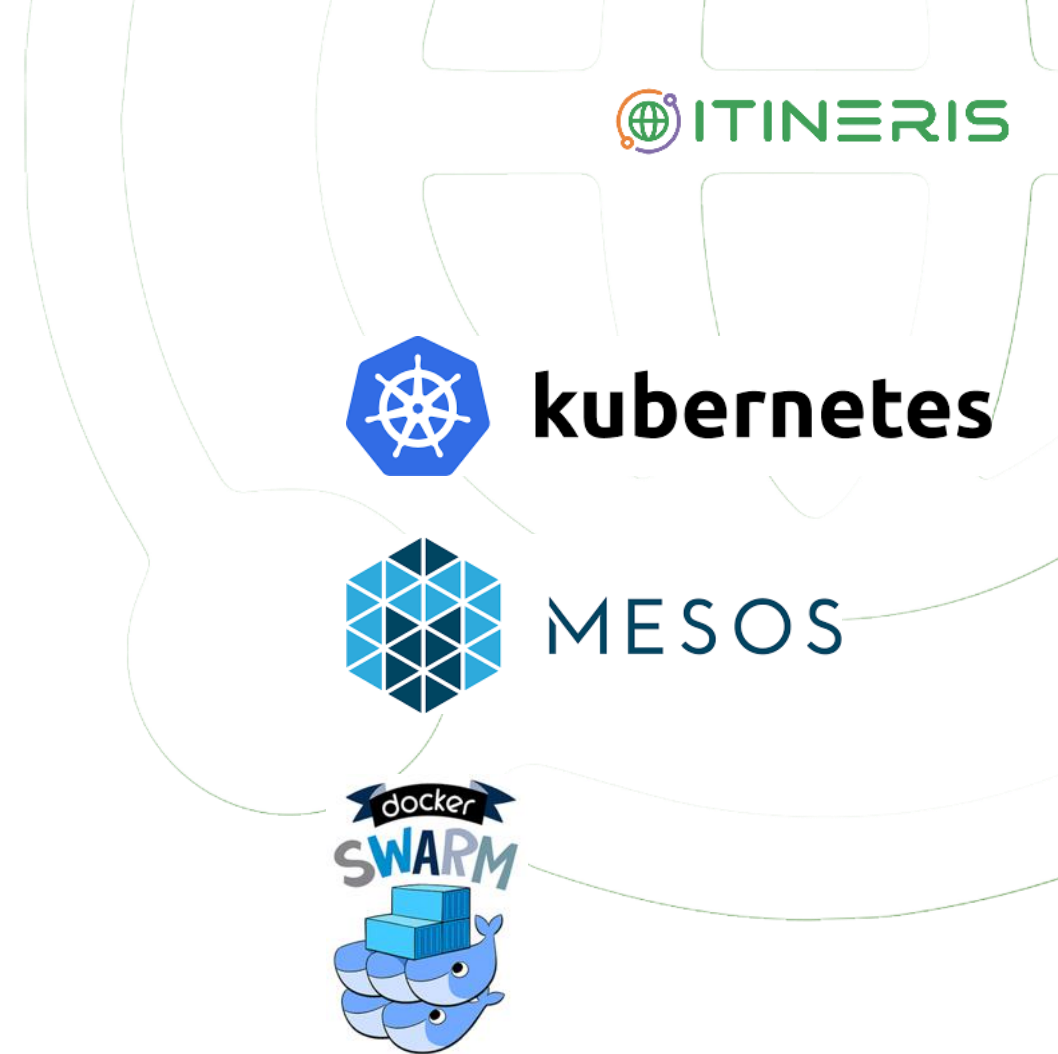


Docker orchestration over a cluster


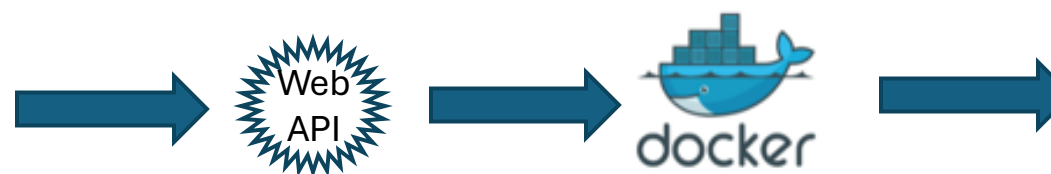
🌐 SWARM: using compose file

🌐 MESOS: using marathon

🌐 Kubernetes: google

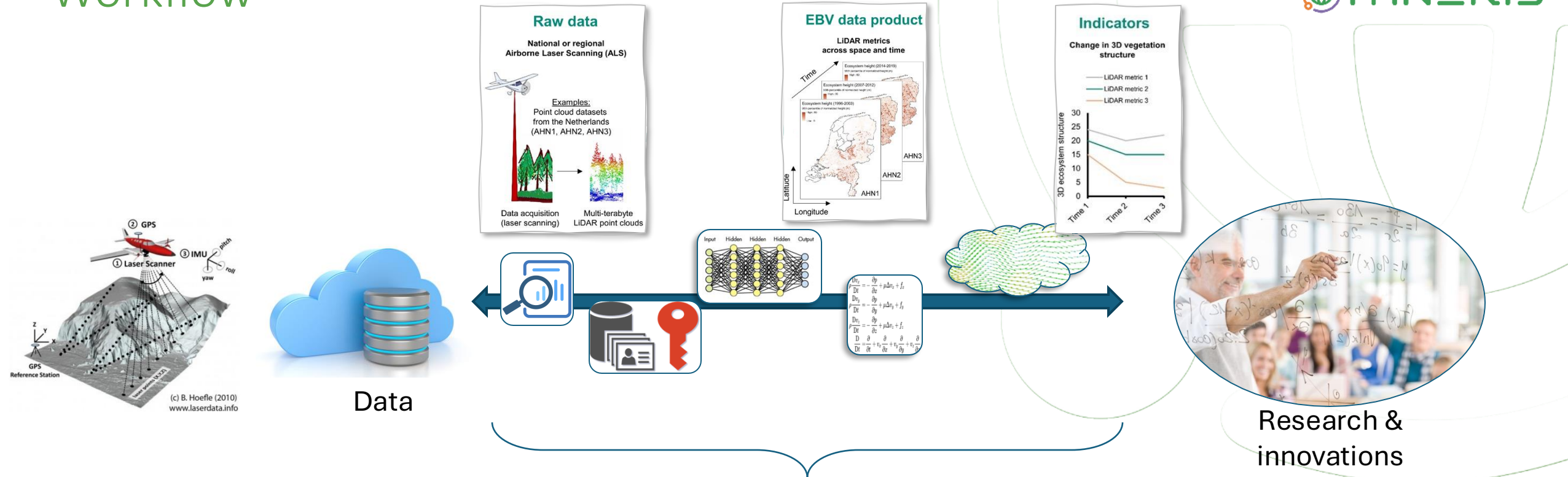


- 
- # Segment (Cell) as a RESTful



Cloud?

Workflow



Tools, algorithms, code, infrastructures and support are often provided by different parties as **services via **internet**.**

Discussion

 What are the benefits for using Cloud?

Discussion



THANKS!

IR0000032 – ITINERIS, Italian Integrated Environmental Research Infrastructures System
(D.D. n. 130/2022 - CUP B53C22002150006) Funded by EU - Next Generation EU PNRR-
Mission 4 "Education and Research" - Component 2: "From research to business" - Investment
3.1: "Fund for the realisation of an integrated system of research and innovation infrastructures"



Finanziato
dall'Unione europea
NextGenerationEU



Ministero
dell'Università
e della Ricerca



Italiadomani
INIZIATIVA NAZIONALE
PER IL FUTURO

