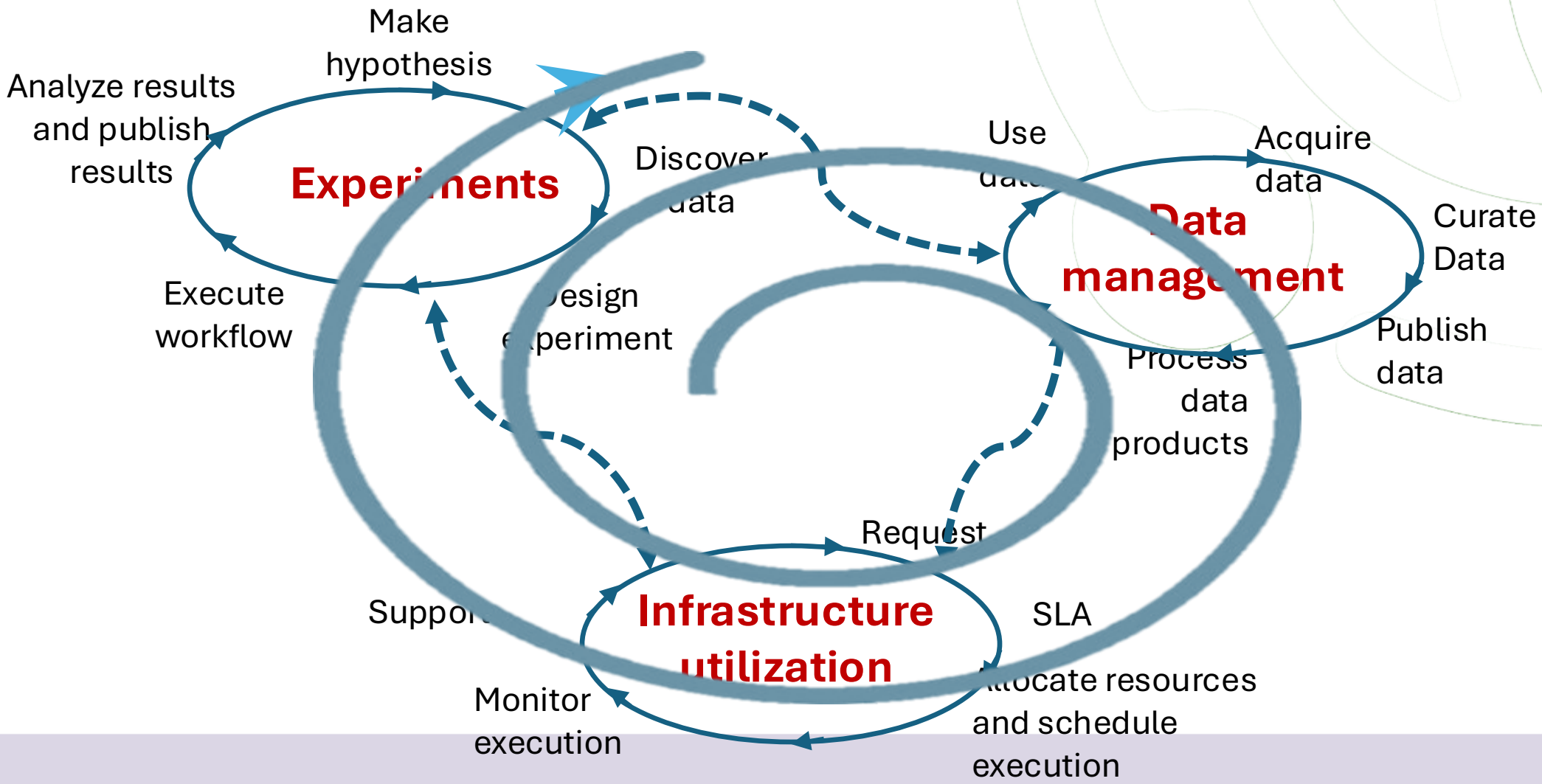
Data management and cloud computing

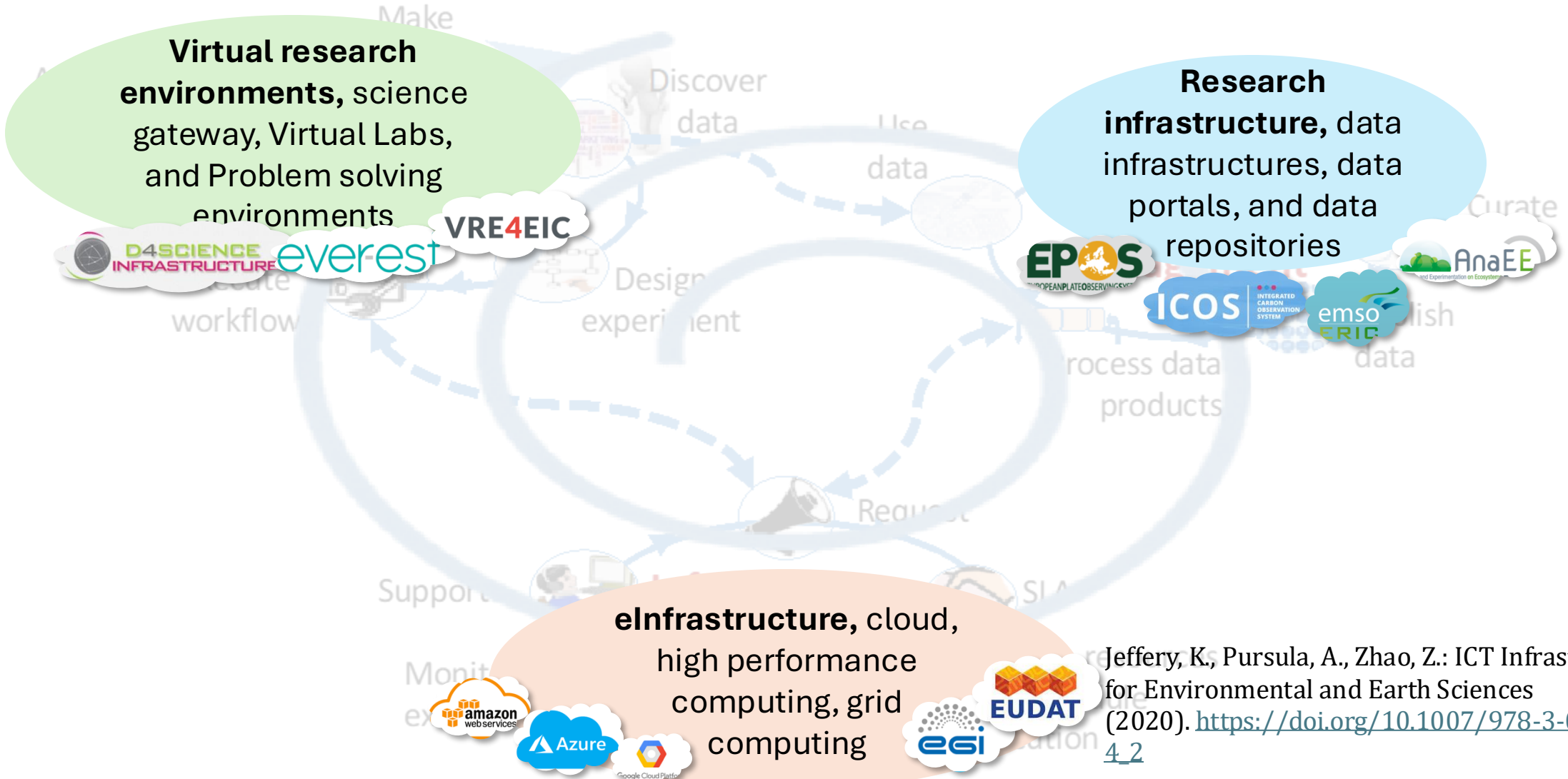
Zhiming Zhao
University of Amsterdam,
LifeWatch Virtual Lab & Innovation Center (VLIC)

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



Jeffery, K., Pursula, A., Zhao, Z.: ICT Infrastructures for Environmental and Earth Sciences (2020). https://doi.org/10.1007/978-3-030-52829-4_2

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

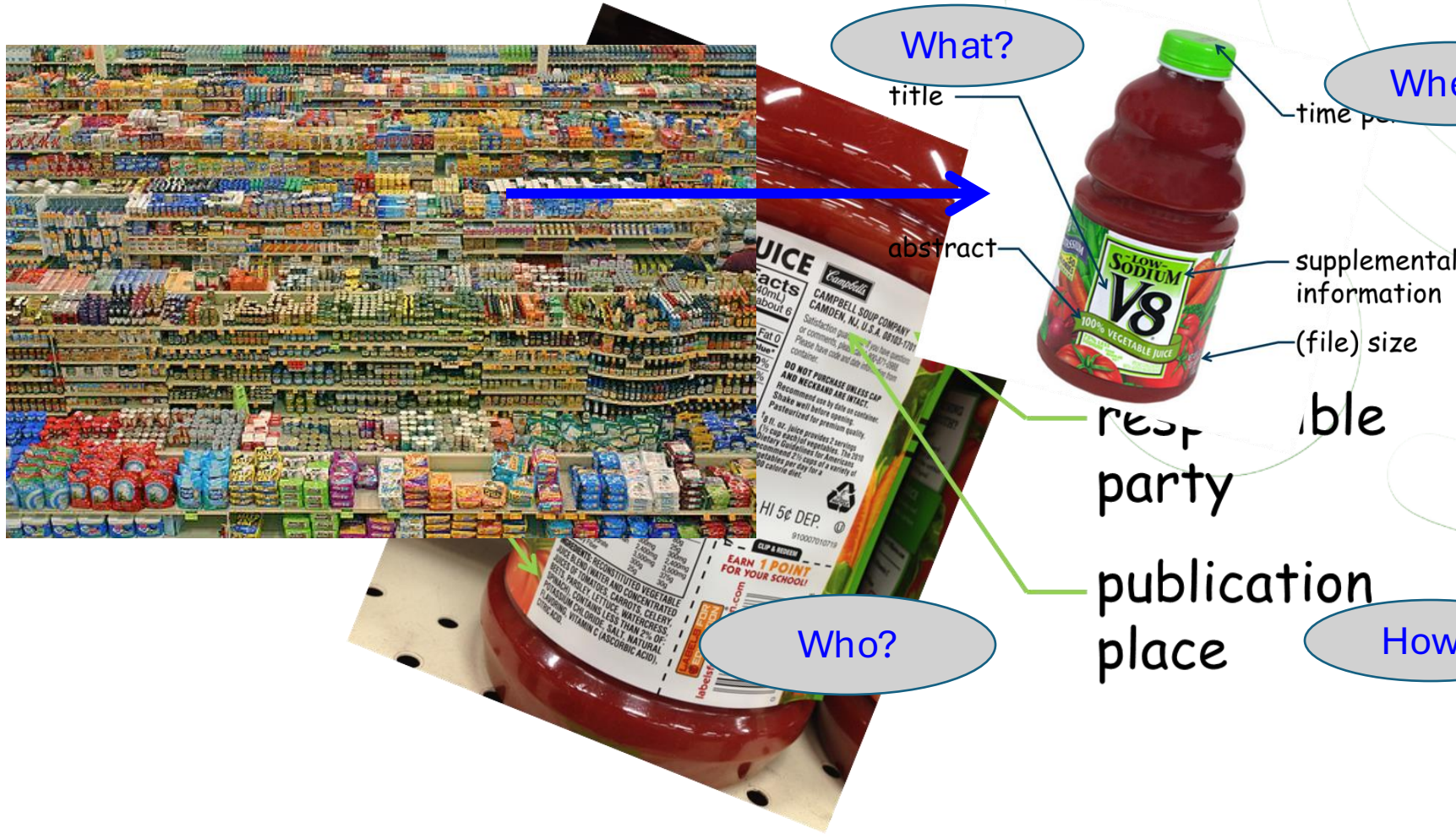
1. Metadata Catalogue

Discussion

- Have you used any data or service catalogues?

6

Metadata for product



What is a catalogue?

- Collections in a Museum,
- Products in shops,
- Services in travel agency
- ….

Catalogue

Directory

Index

Inventory

portfolio

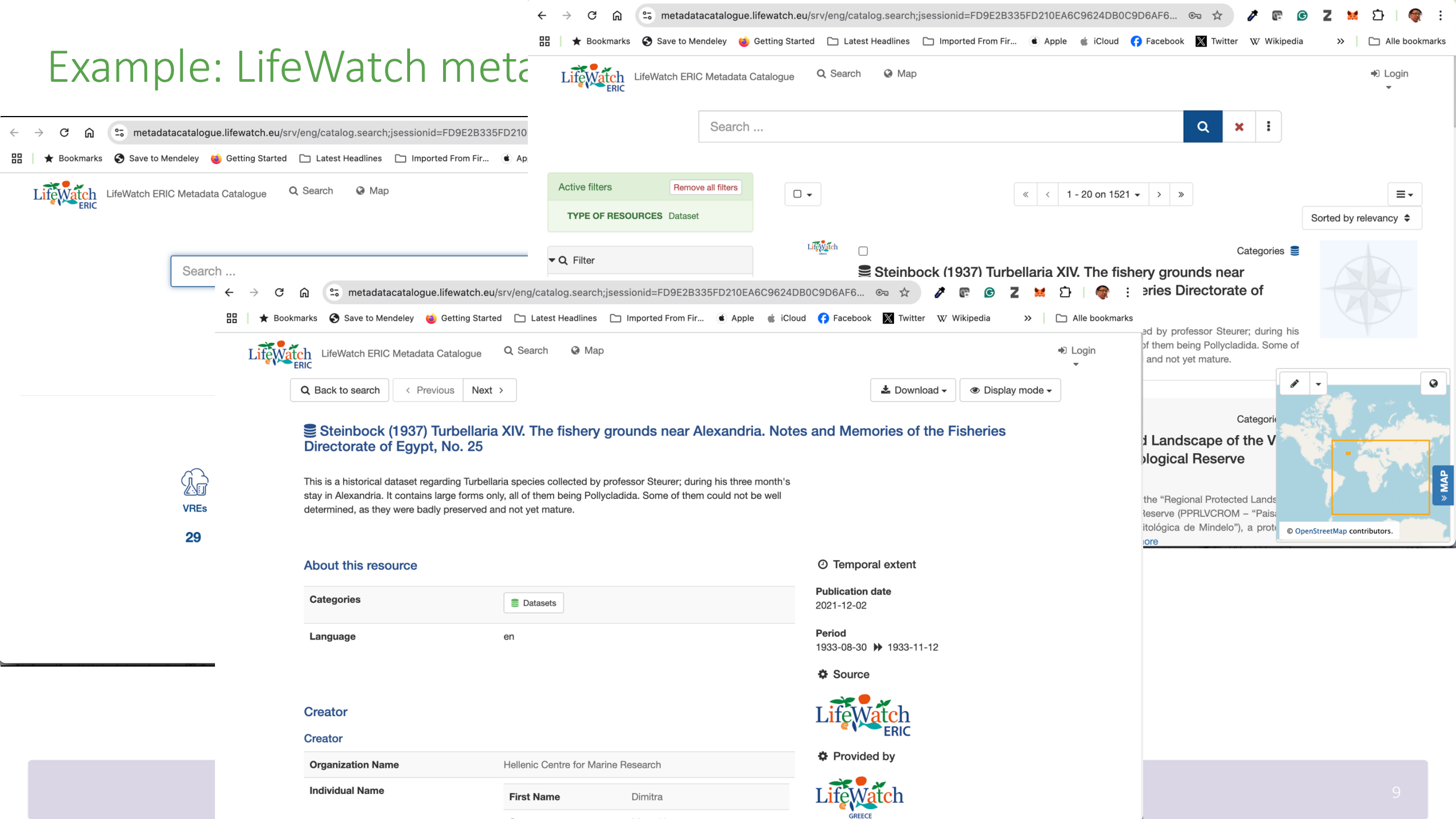
registry

list

….

….

Example: LifeWatch meta

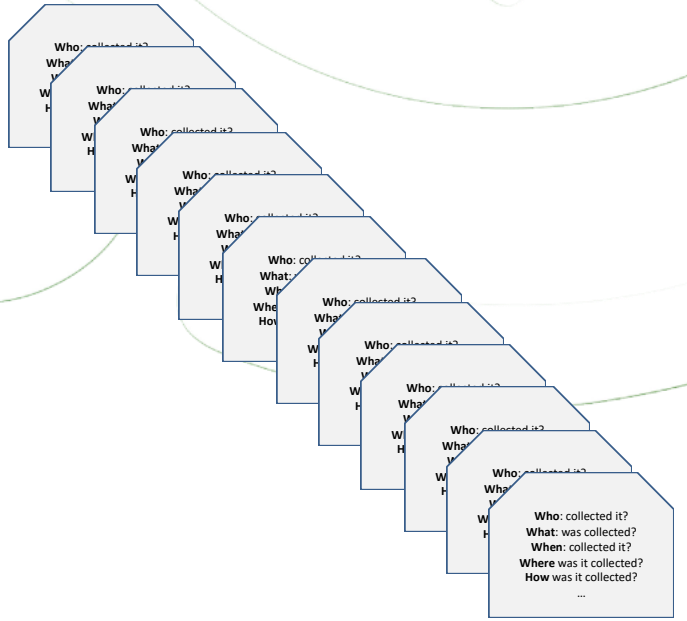


How to make a catalogue?
- simple example

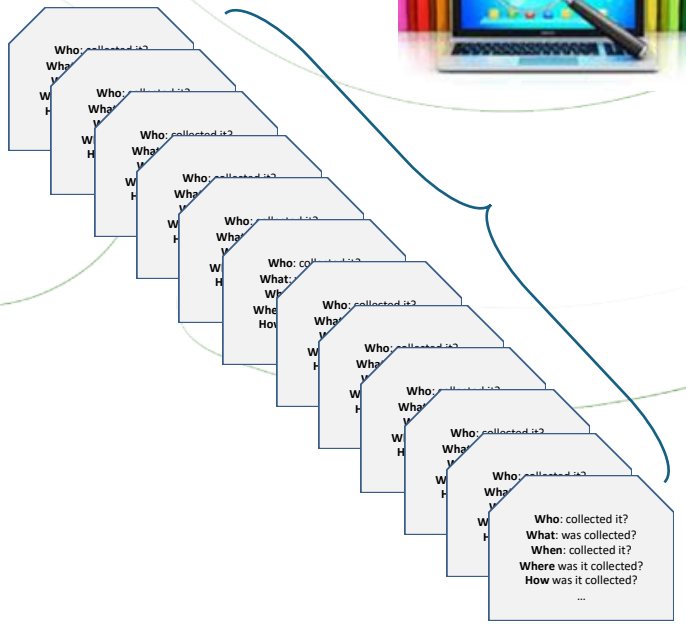
How to make a catalogue?
Step 1: create metadata information of items

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



Technologies

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

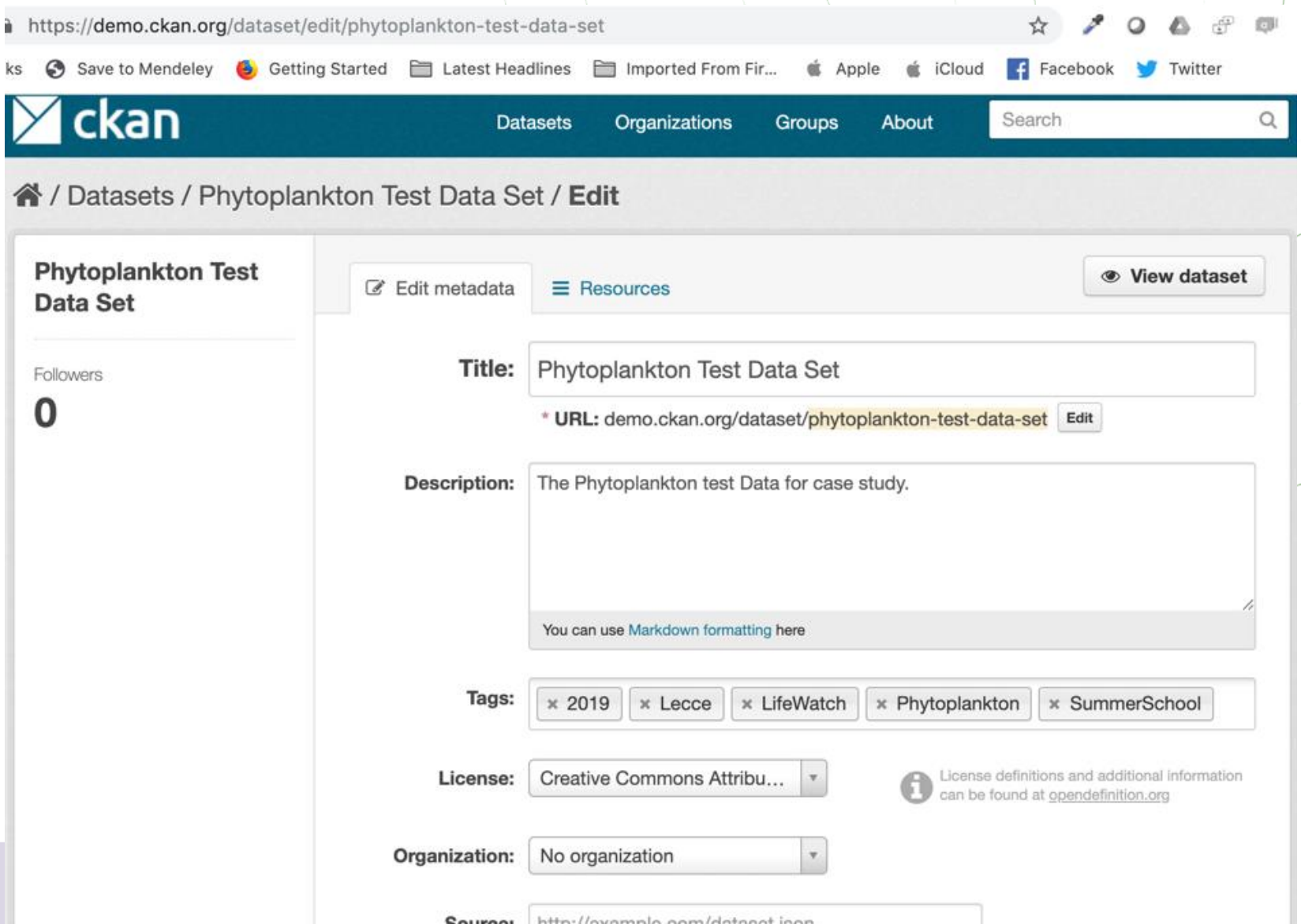
- a web-based open-source management system for the storage and distribution of open data,
- a powerful data catalogue system that is mainly used by public institutions seeking to share their data with the general public.

Open source, Python web app, PostgreSQL DB, GPL

https://demo.ckan.org/

Create data set

- Edit metadata
- Metadata
  - Title
  - Description
  - Tag
  - Organizations
  - License
  - Identifier



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

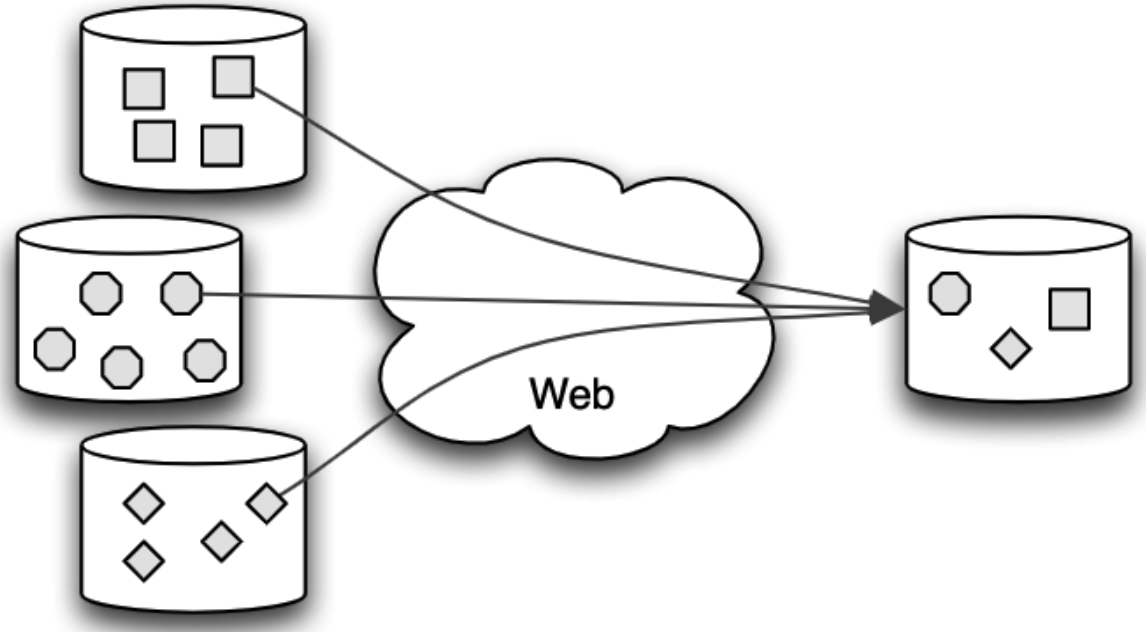
- The GeoNetwork project started out in year 2001 as a **Spatial Data Catalogue System** for the Food and Agriculture organisation of the United Nations (**FAO**), the United Nations World Food Programme (**WFP**) and the United Nations Environmental Programme (**UNEP**).
- At present the project is 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

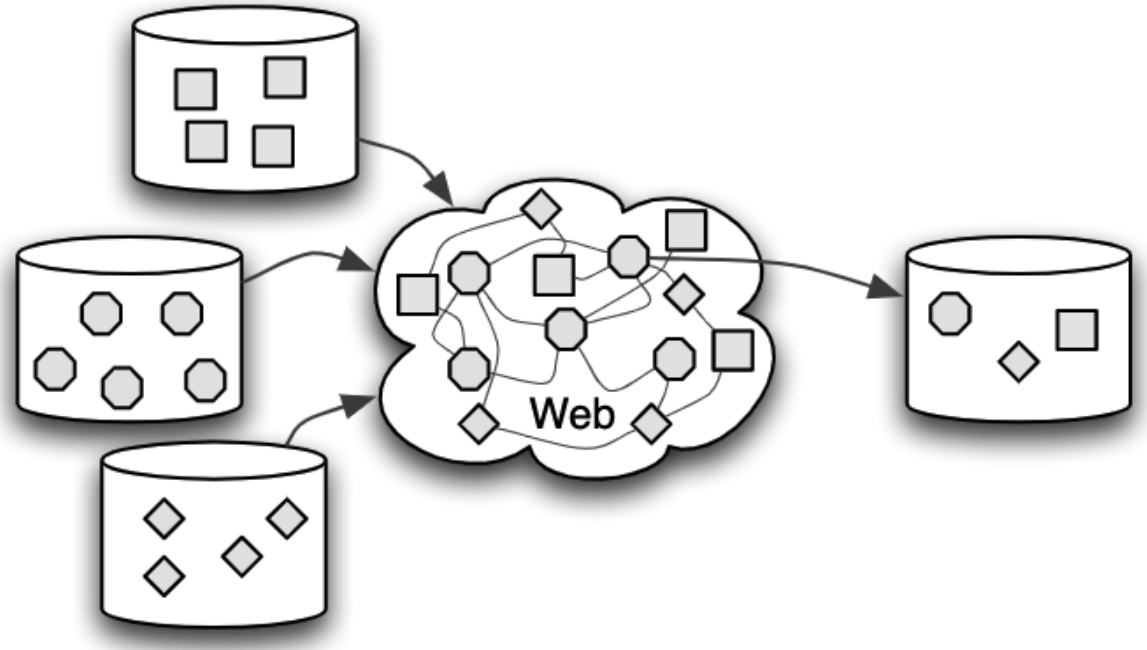
- How does a catalogue enable search?

Other solutions

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

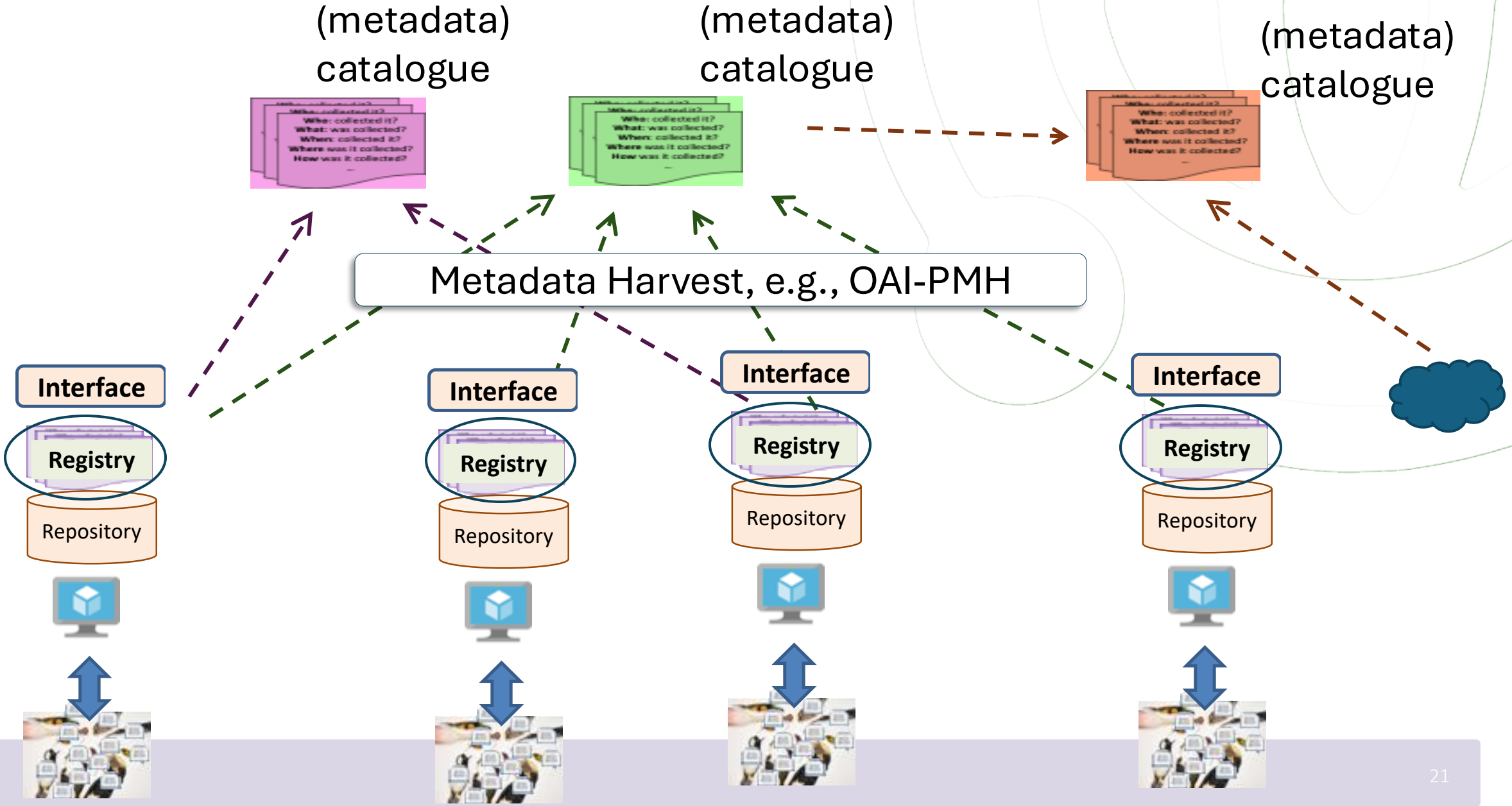


OAI-PMH Approach



LOD Approach

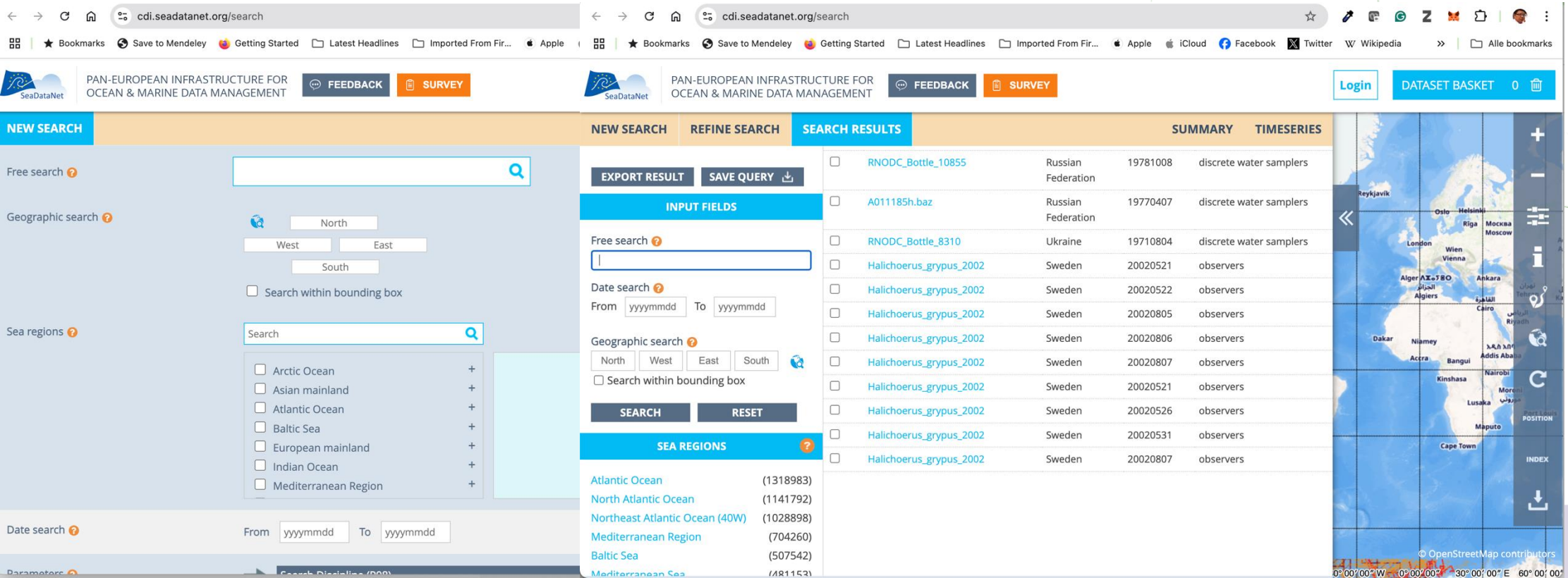
Catalogue for publishing and discovery



Discussion

How does a catalogue enable search?

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



Discussion

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

2. Search engine

Discussion

- What is your search experience?

25

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

26

Natural language analysis

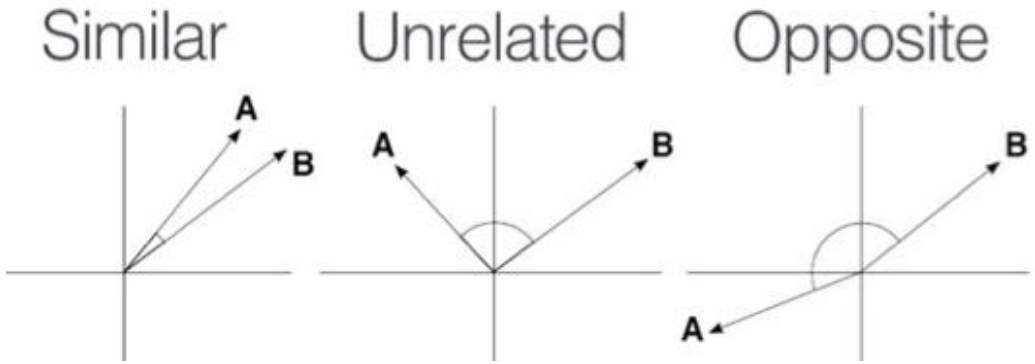
- “bag of words” in text search
  - if search {“biodiversity”, “digital”, “twin”} from different documents
    - Document 1: contains (“*essential*”, “biodiversity”, “*variables*”)
    - Document 2: contains (“digital”, “samples”)
    - Document 3: contains (“physical”, “twin”)
  - Assumption
    - A document might be relevant if it contains one of the keyword s
    - 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

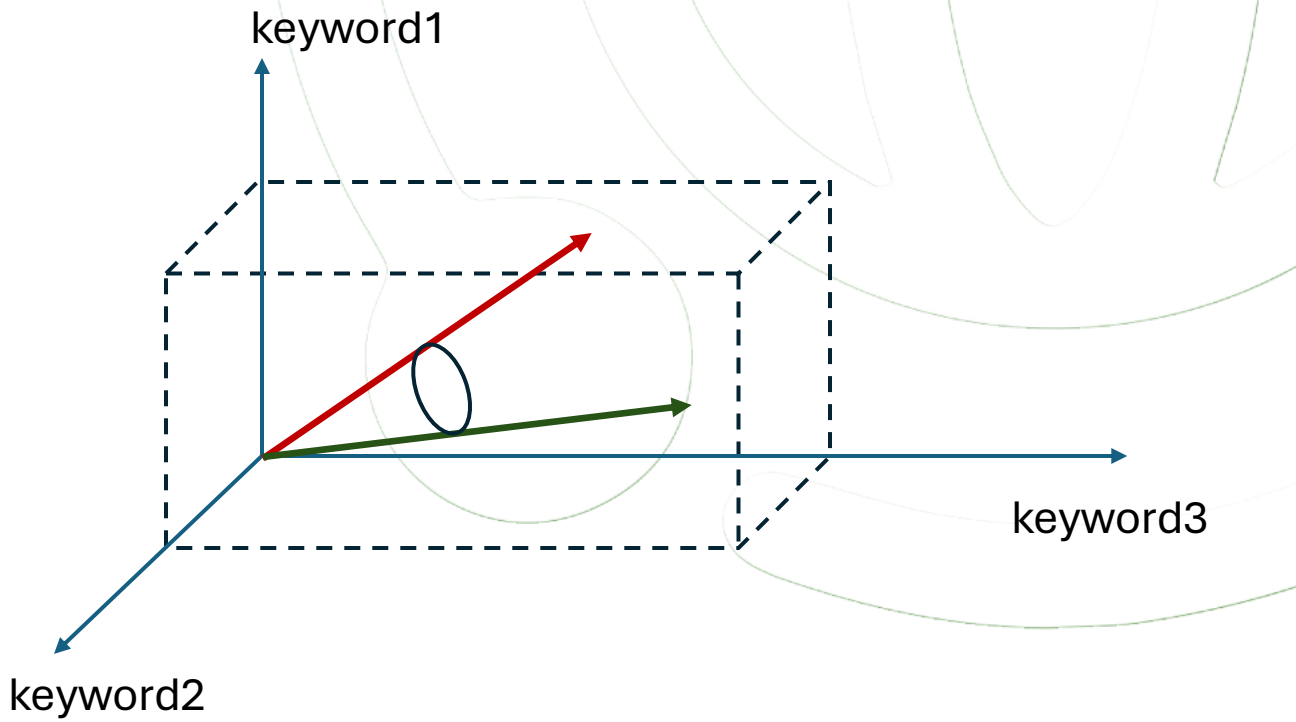
- Represent documents and possible queries as an N-dimensional vector space;
  - Term: basic concepts and words in all documents and queries.
  - Each term defines a dimension in the vector
  - Document vector: (t1, t2, …. tn); ti is document term weight
  - Query vector: (q1, q2, ….qn); qi is query term weight
  - Relevance (q, d)

Similarity

- Cosine similarity between vectors

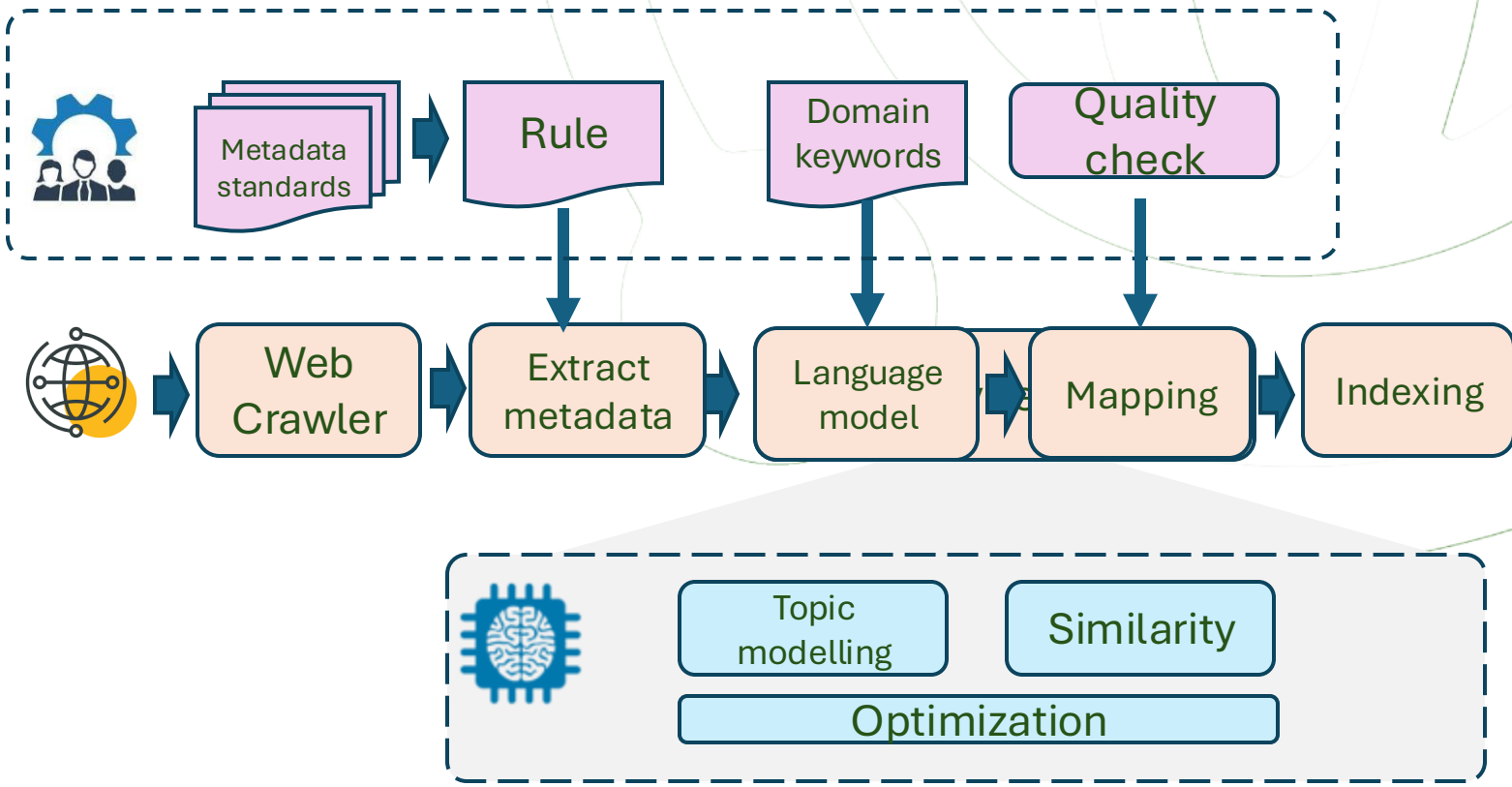


$$\cos\theta = \frac{\vec{a} \cdot \vec{b}}{\|\vec{a}\| \cdot \|\vec{b}\|}$$



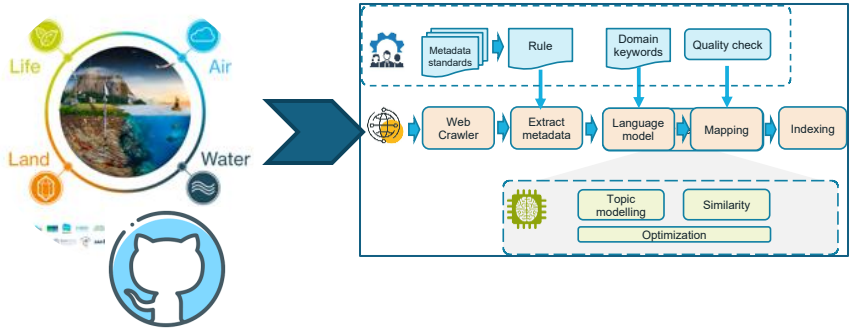
A customizable online information index framework

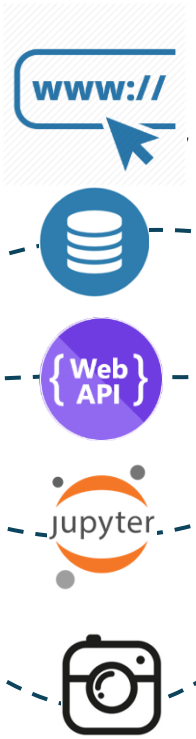
- How to index metadata of research assets?

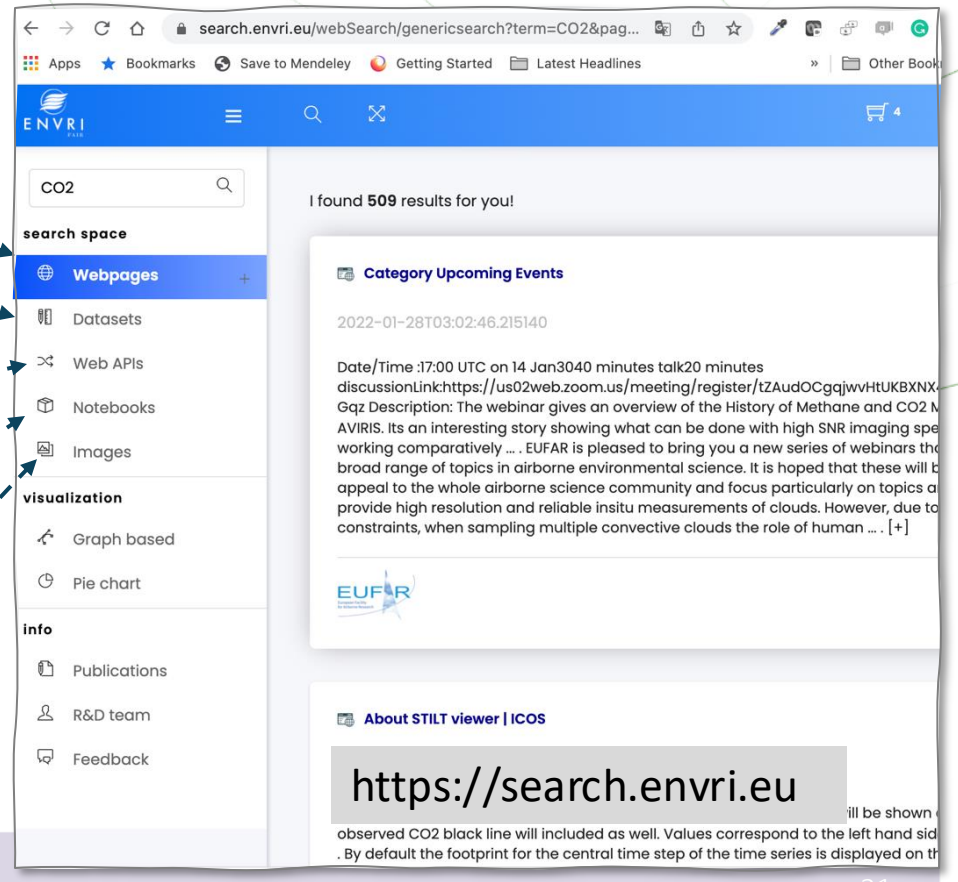


- Index online resources with **insufficient textual information** (e.g., data and API), or **incomplete/inconsistent** metadata information, by
  - **including human experts** in the loop to refine mapping rules and specific domain keywords
  - **extending** the topic/keywords-based information retrieval pipeline for online resources
  - **Using natural language processing techniques** for modelling topics, checking similarities, and optimizing mapping quality

Search engine for different resource types



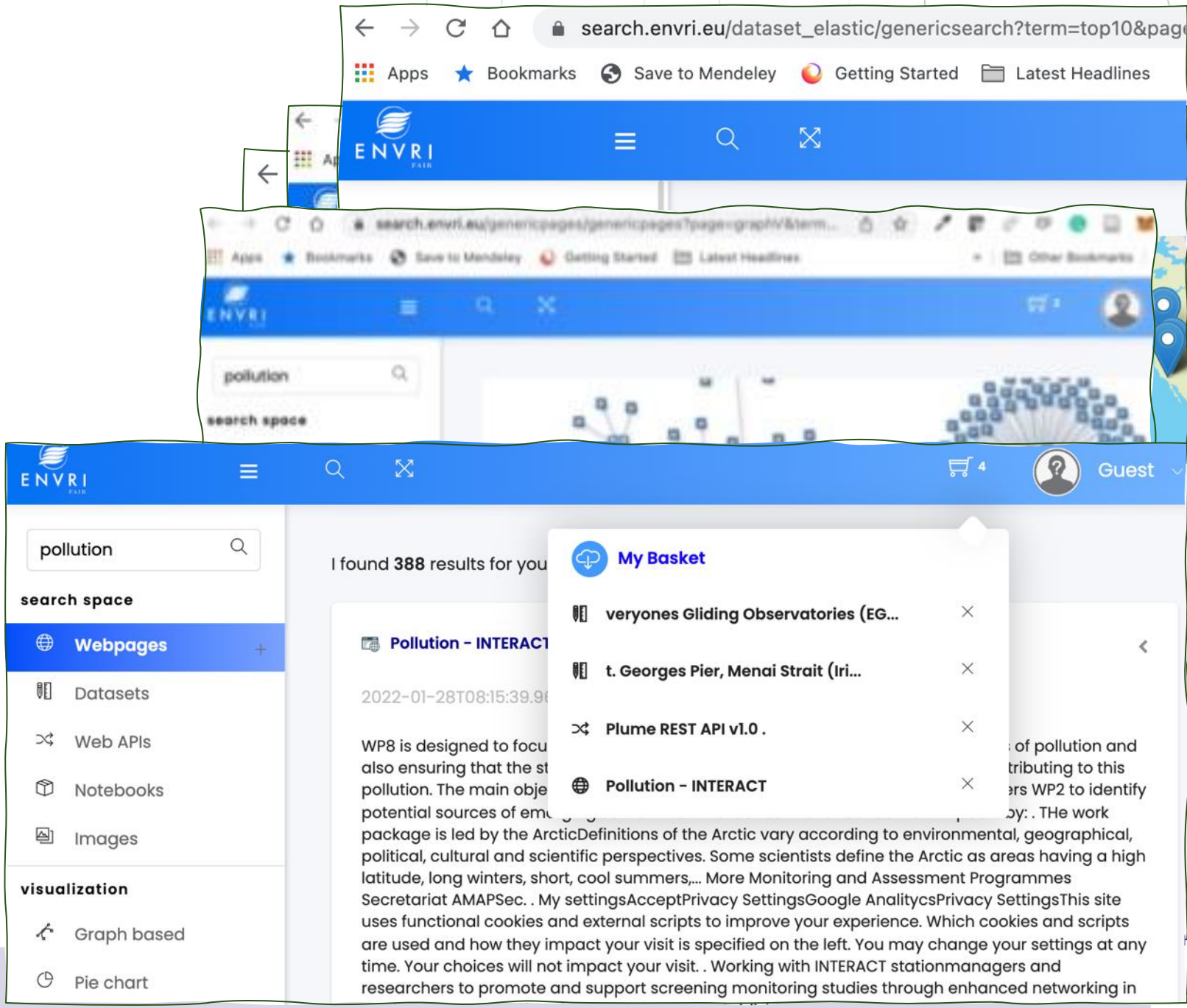




- Indexed all online sources (obtained from the FAIR assessments)
  Web pages, data sets, APIs, Notebooks, Images ...

ENVRI Knowledgebase

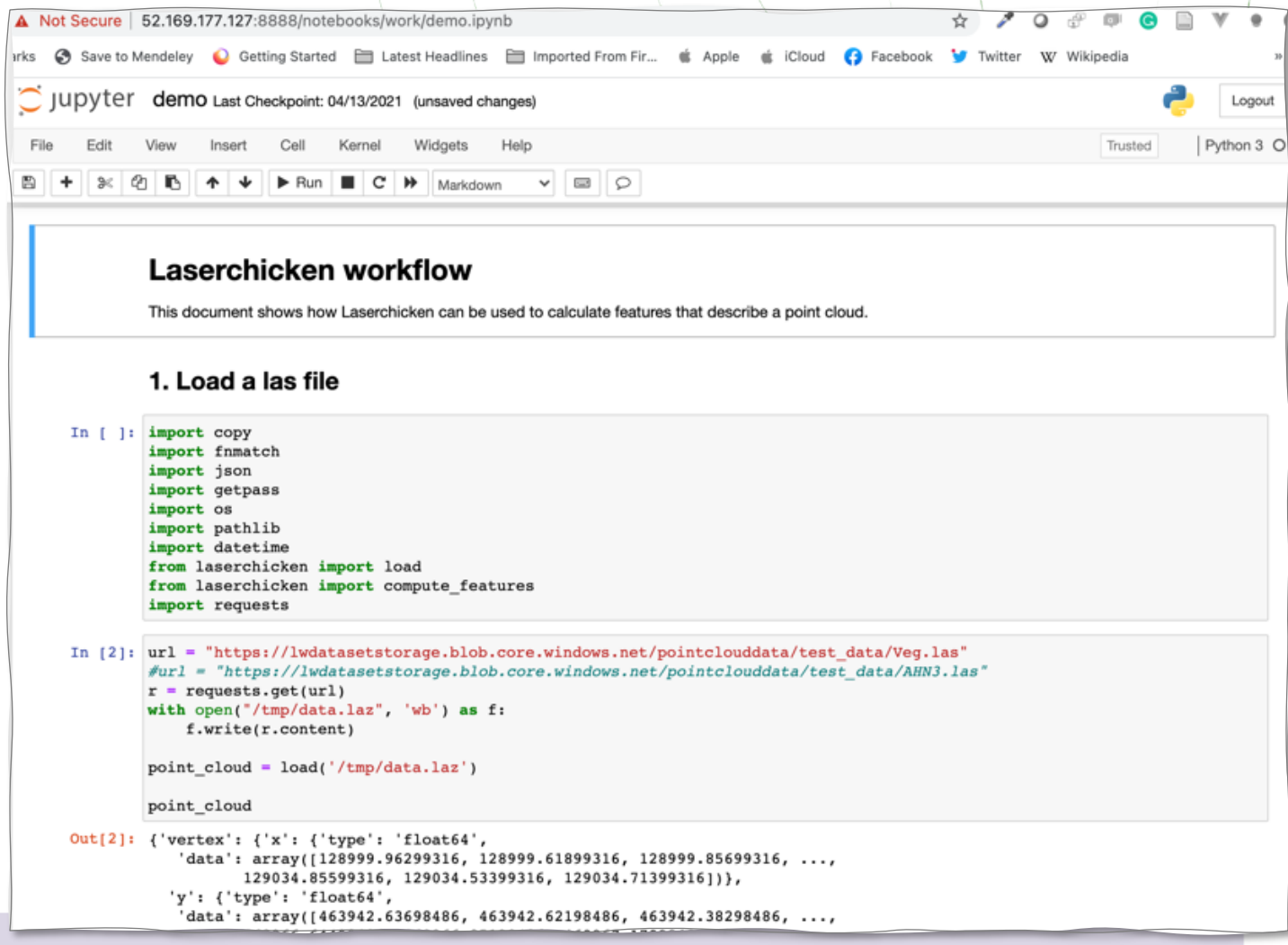
- Knowledge base interface adopts the design style of ENVRI-HUB
- Category online resources as: web pages, data sets, Notebook, API, images, (more to come)
- Content filtering
- Better visualization
- Concept of basket



1. The ENVRI-KMS should include all prospective RIs, datasets, repositories, best practices...

Search jupter 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.)



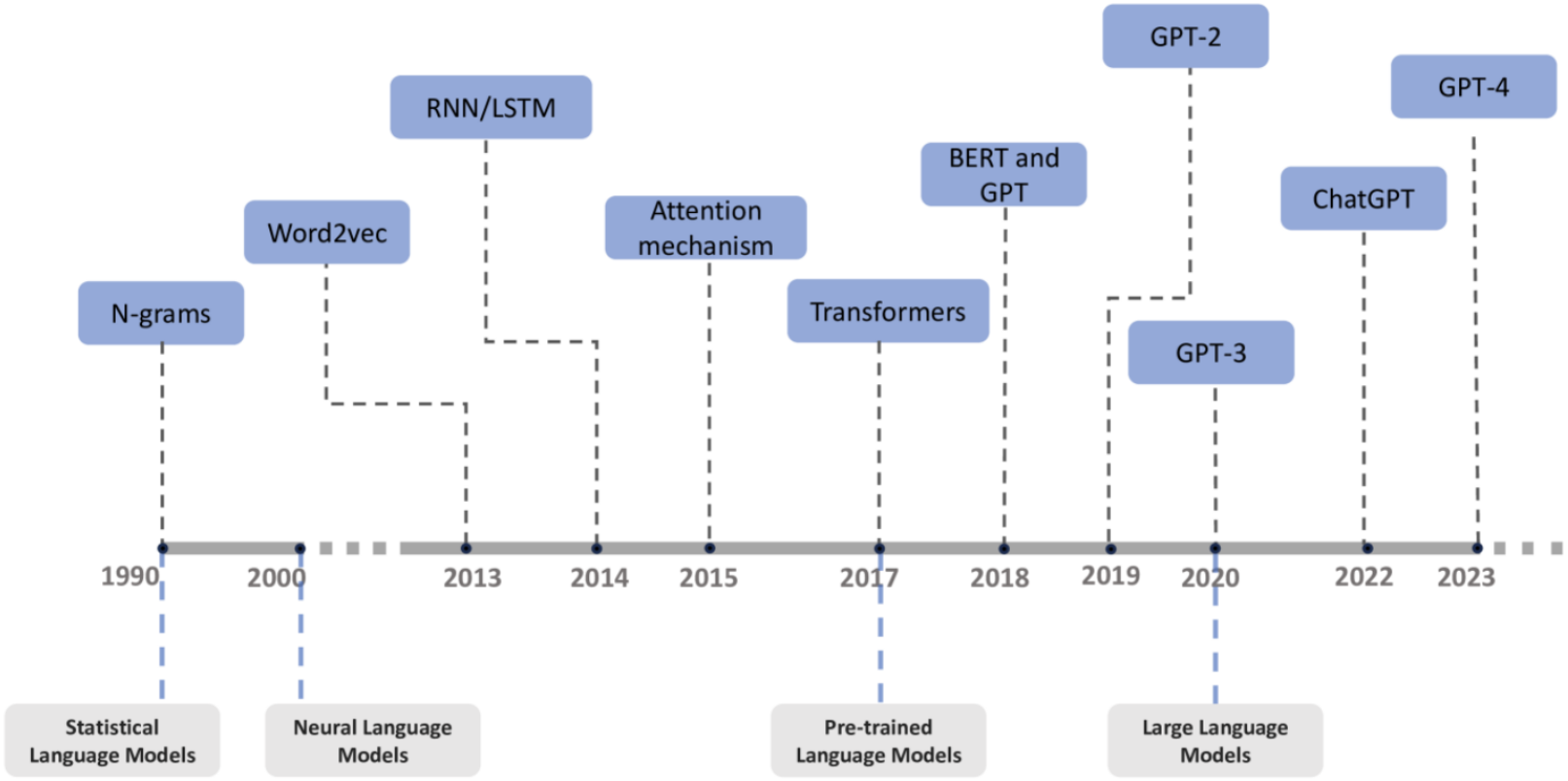
Discussion: how to improve the search quality?

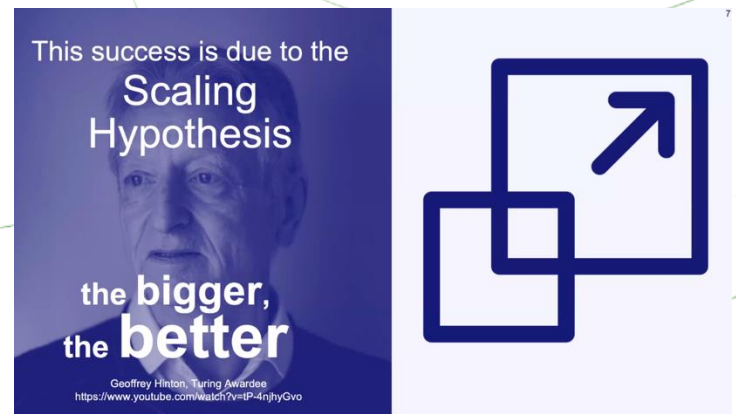
- How to better rank the relevance?
- How to “guess” the intent of the user?
- …

34

Discussion: Search using Advanced AI

- Large language model (LLM)

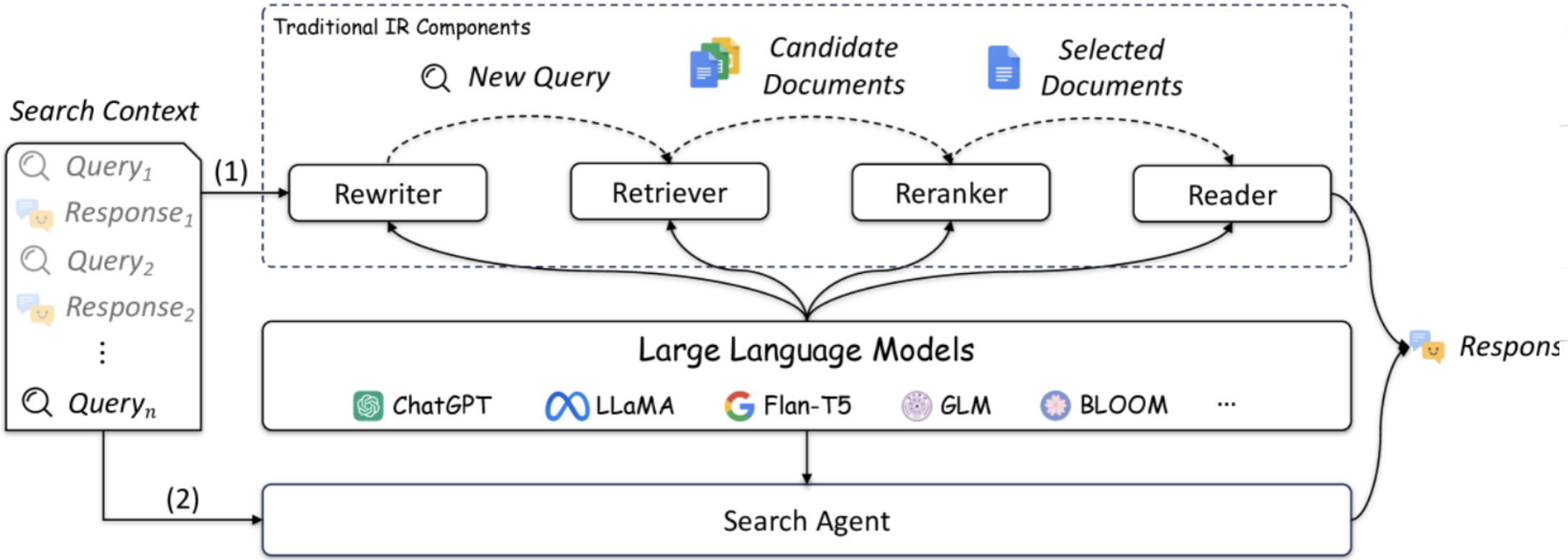




35

Discussion: LLM in search

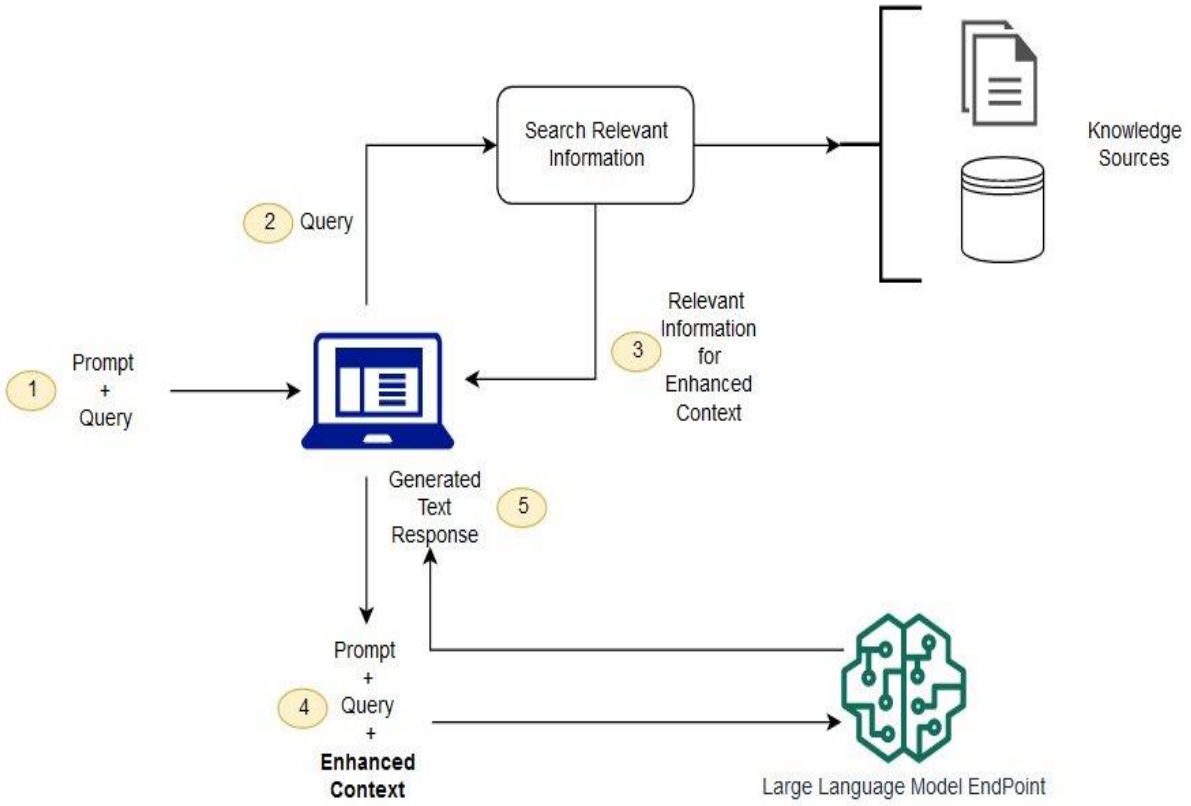
LLM in different search phases



36

Discussion: LLM in search

Retrieval – Augmented Generation (RAG)



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

37

Discussion: Semantic search

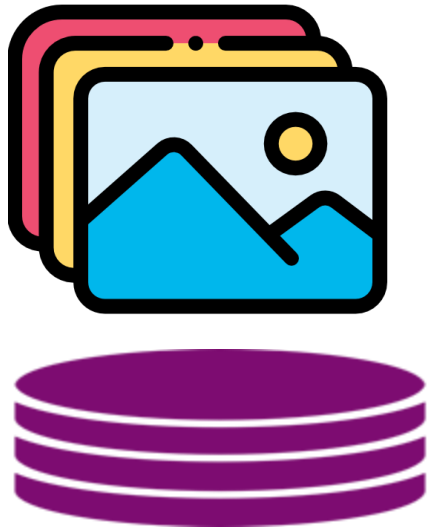
3. Parallel and distributed computing

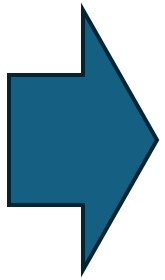
Discussion: requirements for computing and data processing

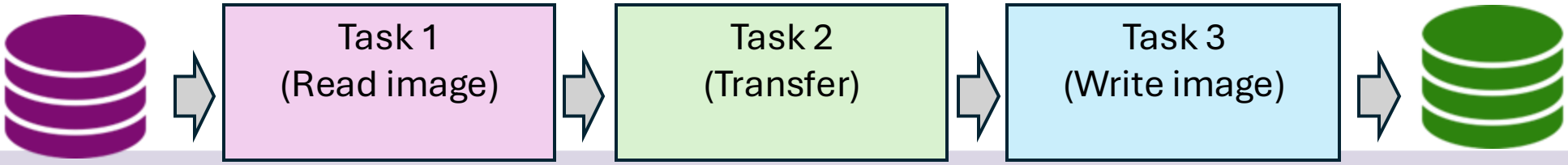
- Requirements

40

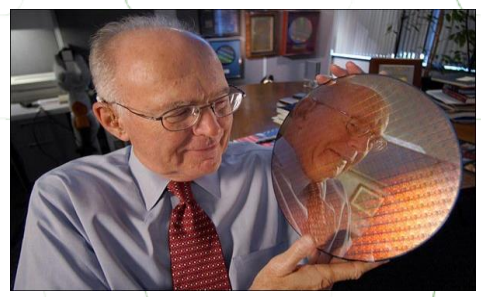
Example 1: unify different image files





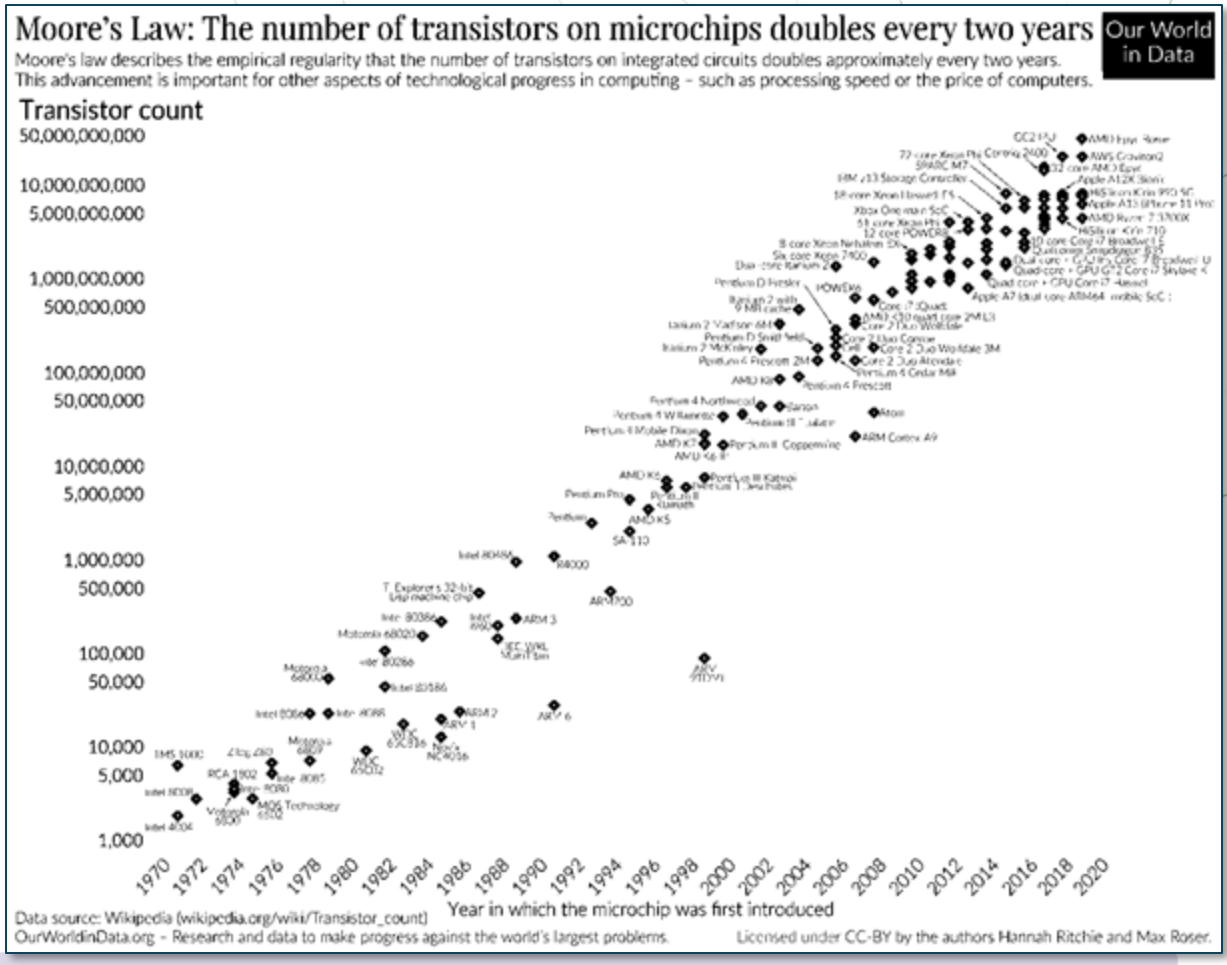


Moore's Law



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

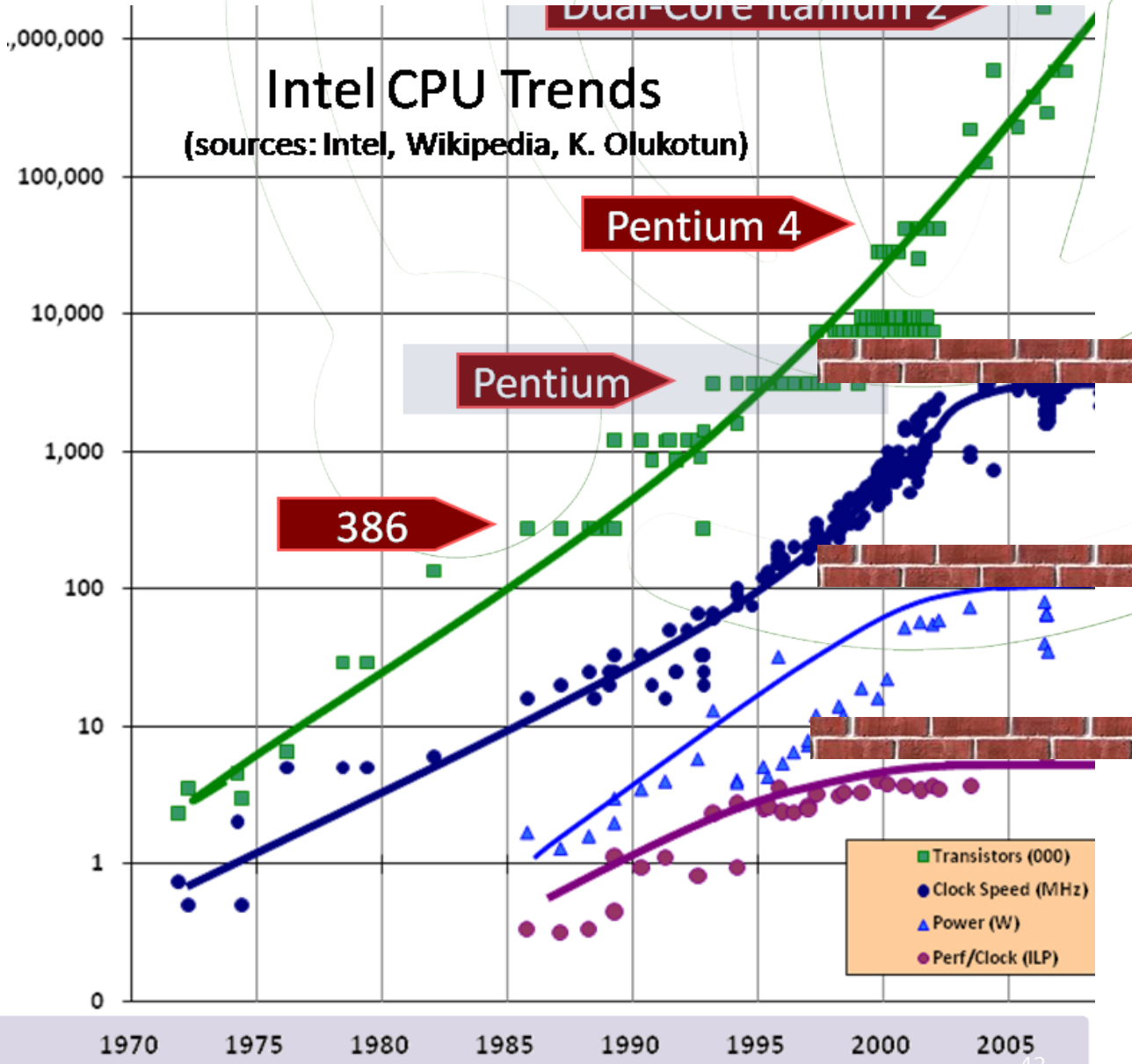
- More **transistors/gates** in a chip
- Higher clock **frequency**
- More advanced **design**



https://ourworldindata.org/uploads/2020/11/Transistor-Count-over-time.pr

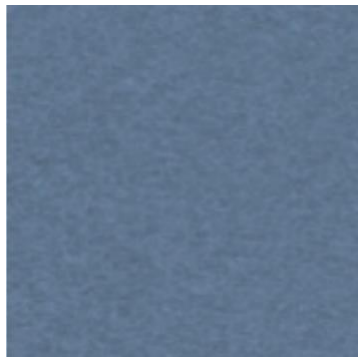
Around 2005: hitting thewalls

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

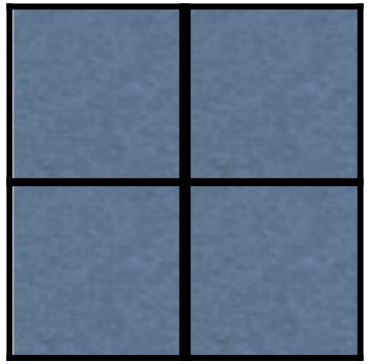


The shift to multi-core

Performance 1
Power 1

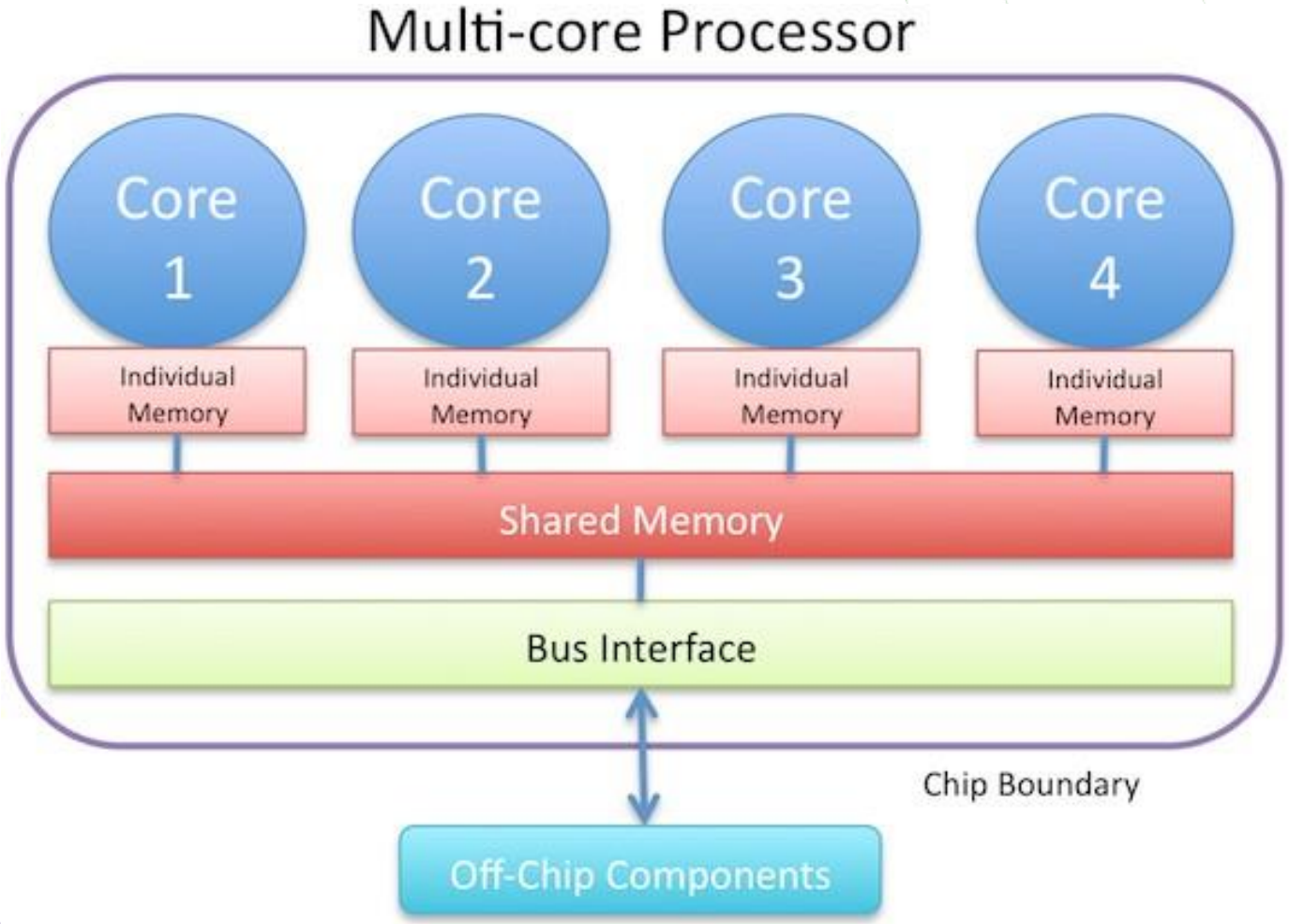


Performance: X **2** (2*F)
Power: X **4**

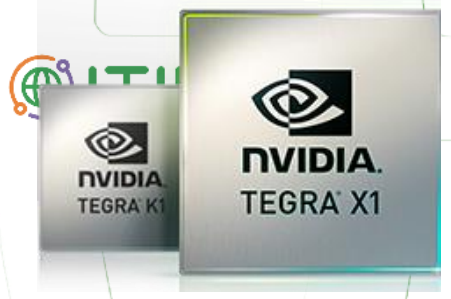


F remains same
Performance = ½ * 4 = 2
Power = ¼ * 4 =1

Generic multi-core CPU

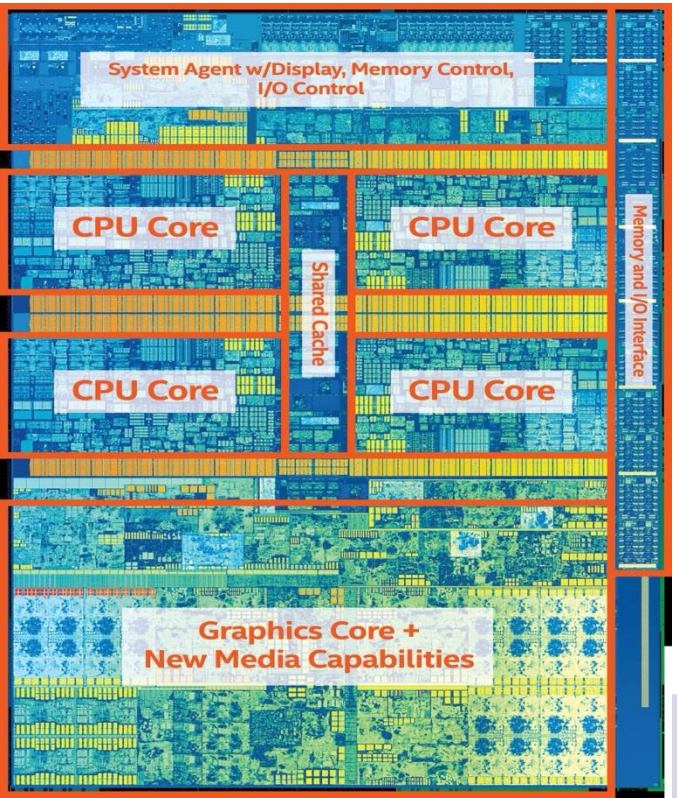


Has been widely used

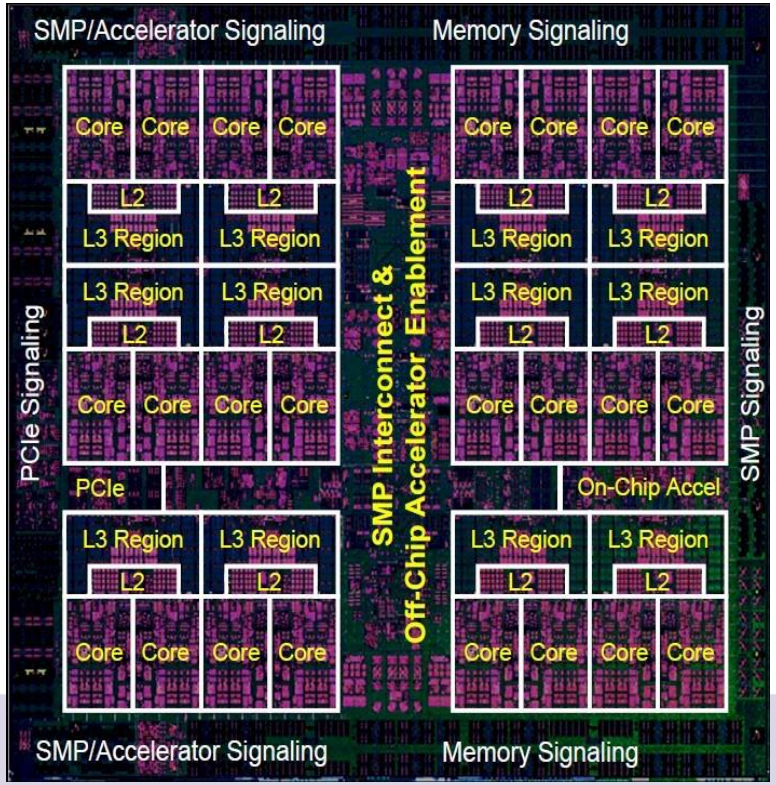


Nvidia Tegra:
Quad-core CPU, 256-core GPU

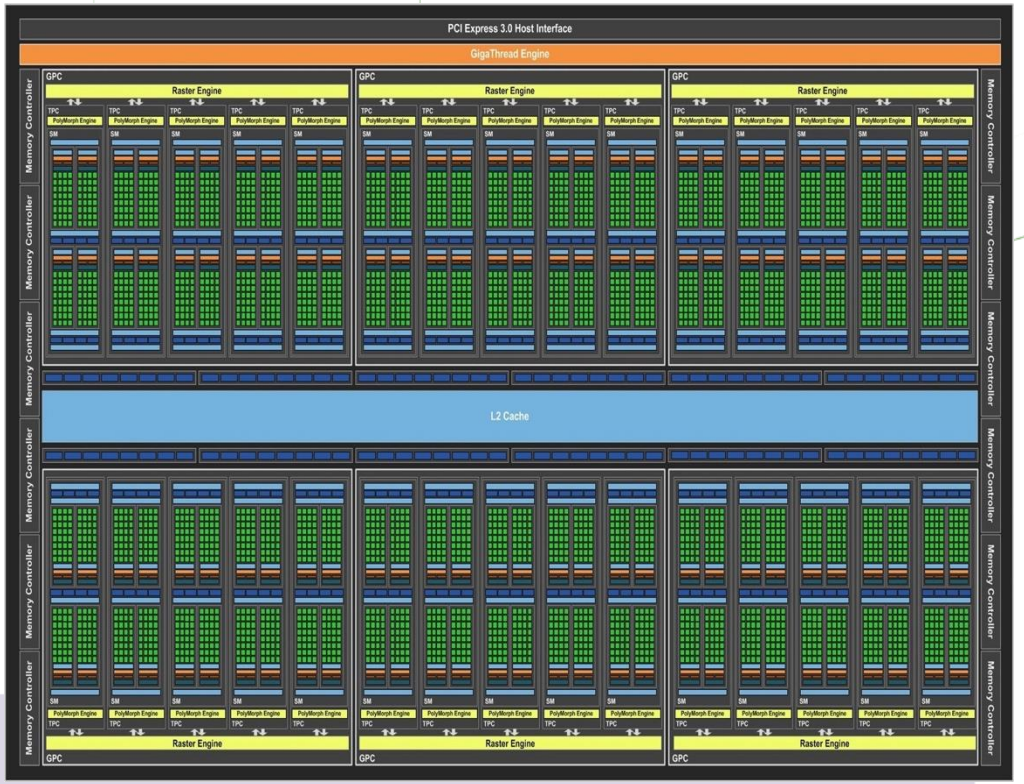
Intel Kaby Lake(2017)



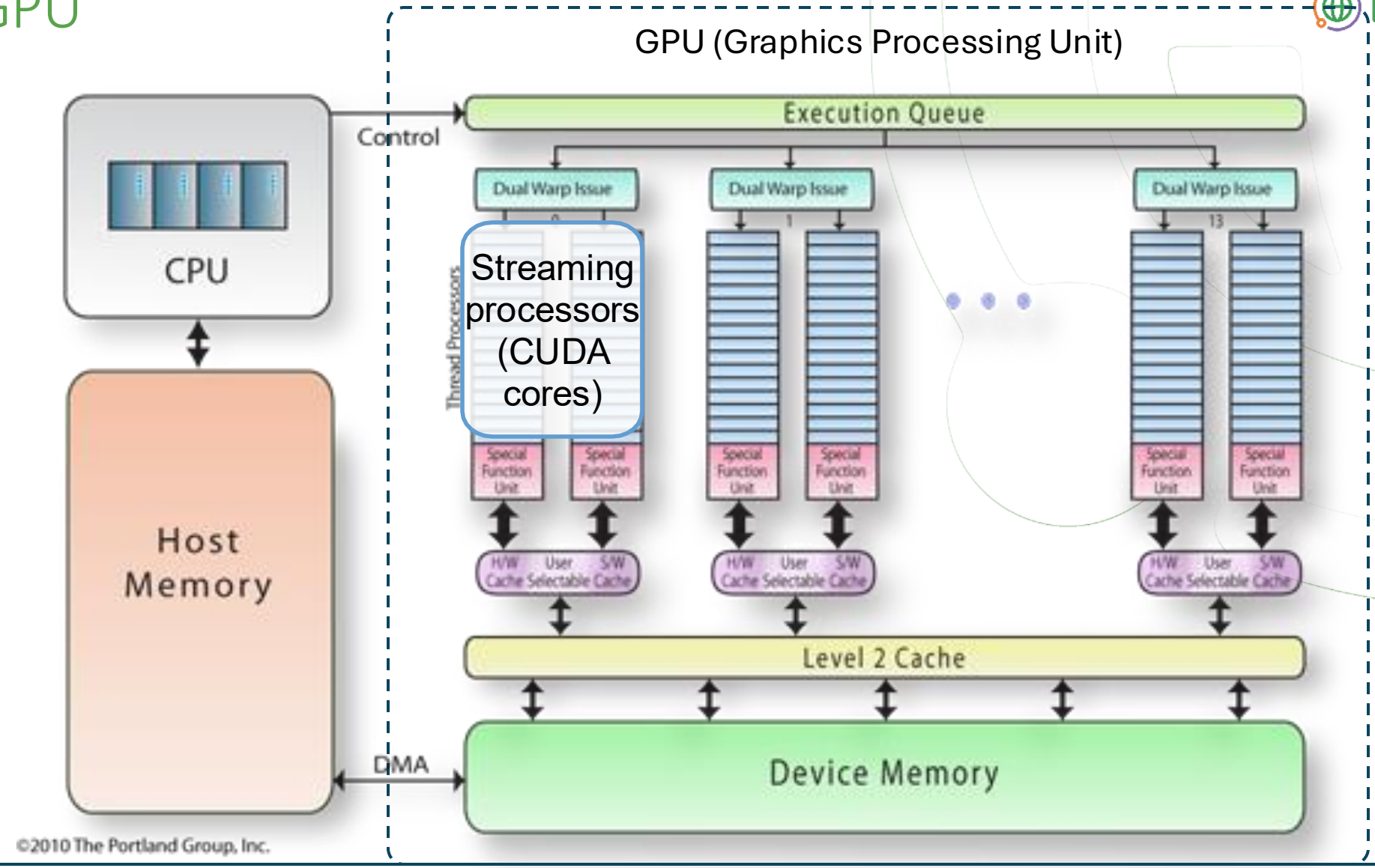
IBM Power9(2016)



Nvidia Titan Xp(2017)



Generic GPU



Streaming processor (SP)

CUDA: computer unified device architecture

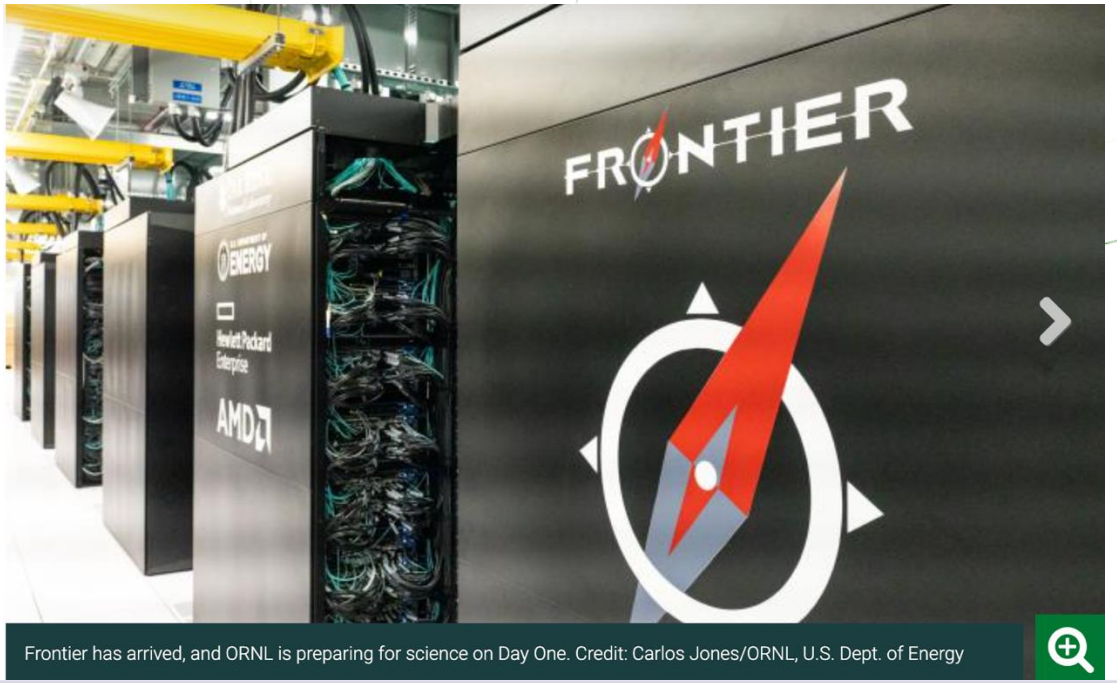
Discussion: why parallelism?

ITINERIS

- 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

https://www.top500.org/lists/top500/2024/06/

Data parallization

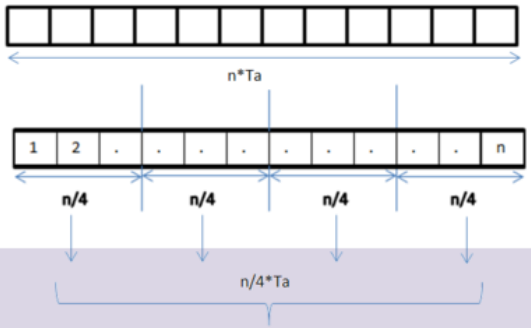
Often used when

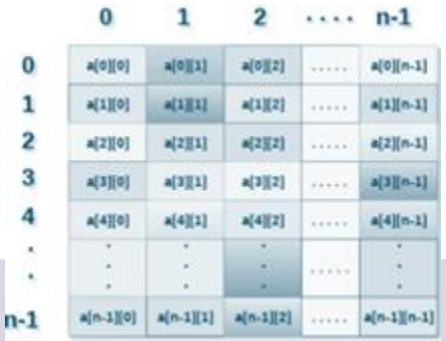
- The application has to handle **a large data input**
- Each of the data points needs to be processed **in the same way** (using the same program)

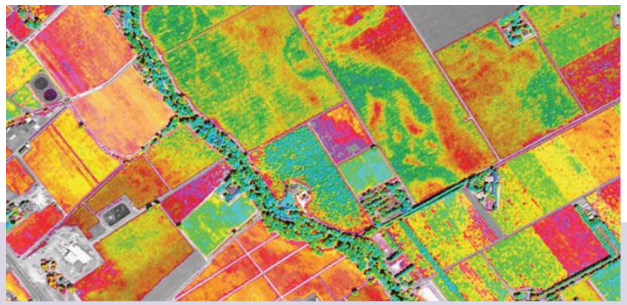
The basic idea

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

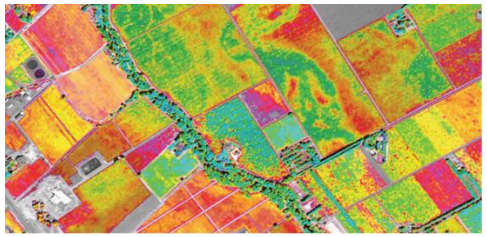
Examples







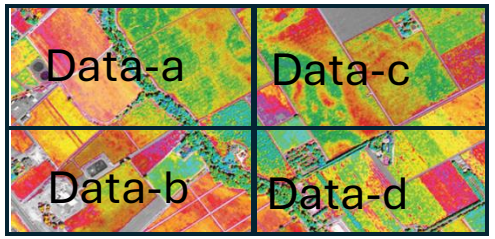
Data parallization



Data

Task 1

Task 2



Data

Task 1a

Task 1b

Task 1c

Task 1d

Task 2

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?

54

Cloud: services, technologies ….

What is Cloud?

Services of resources

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

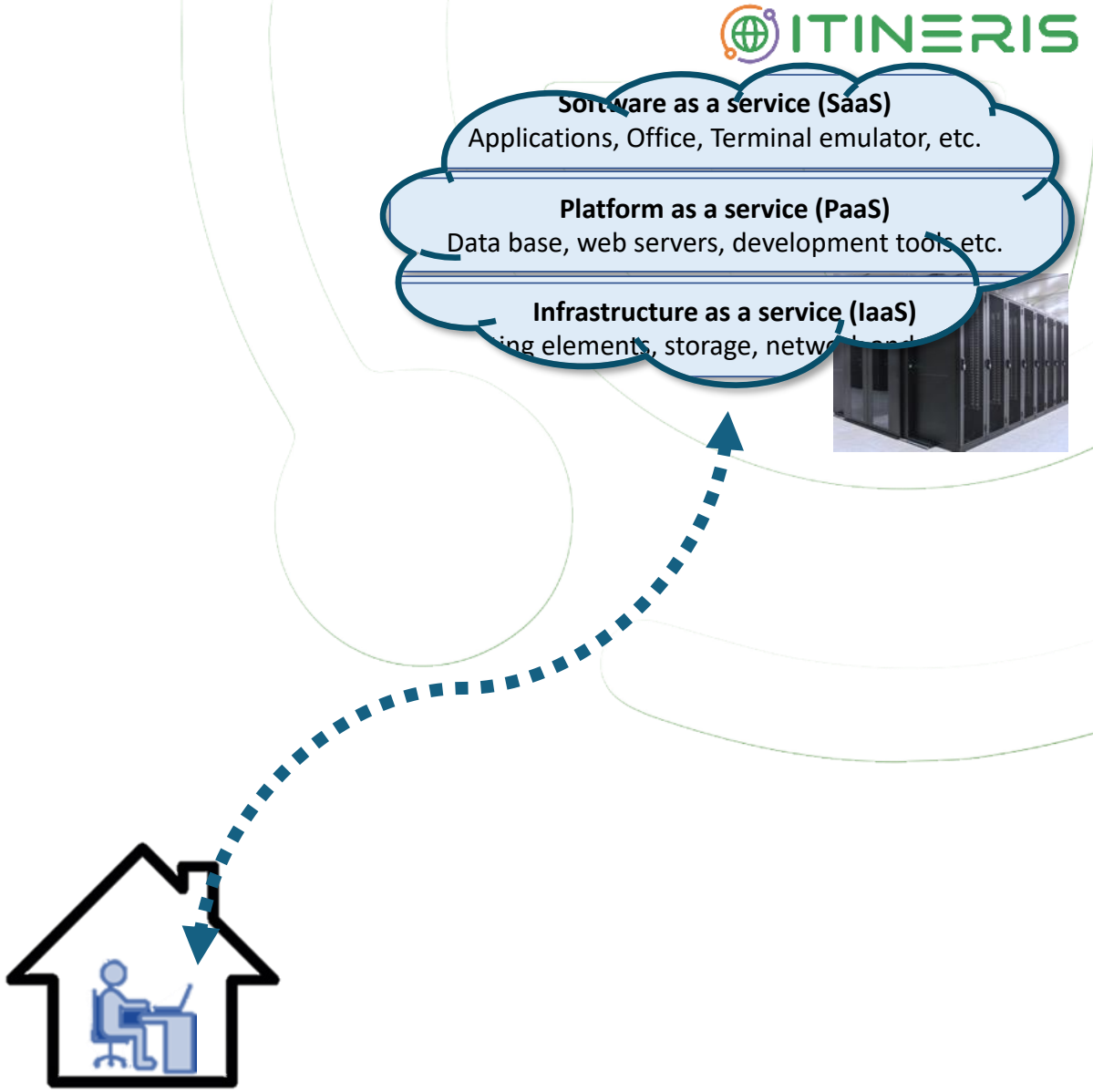
Software as a service (SaaS)
Applications, Office, Terminal emulator, etc.

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

Infrastructure as a service (IaaS)
Computing elements, storage, network and server etc.

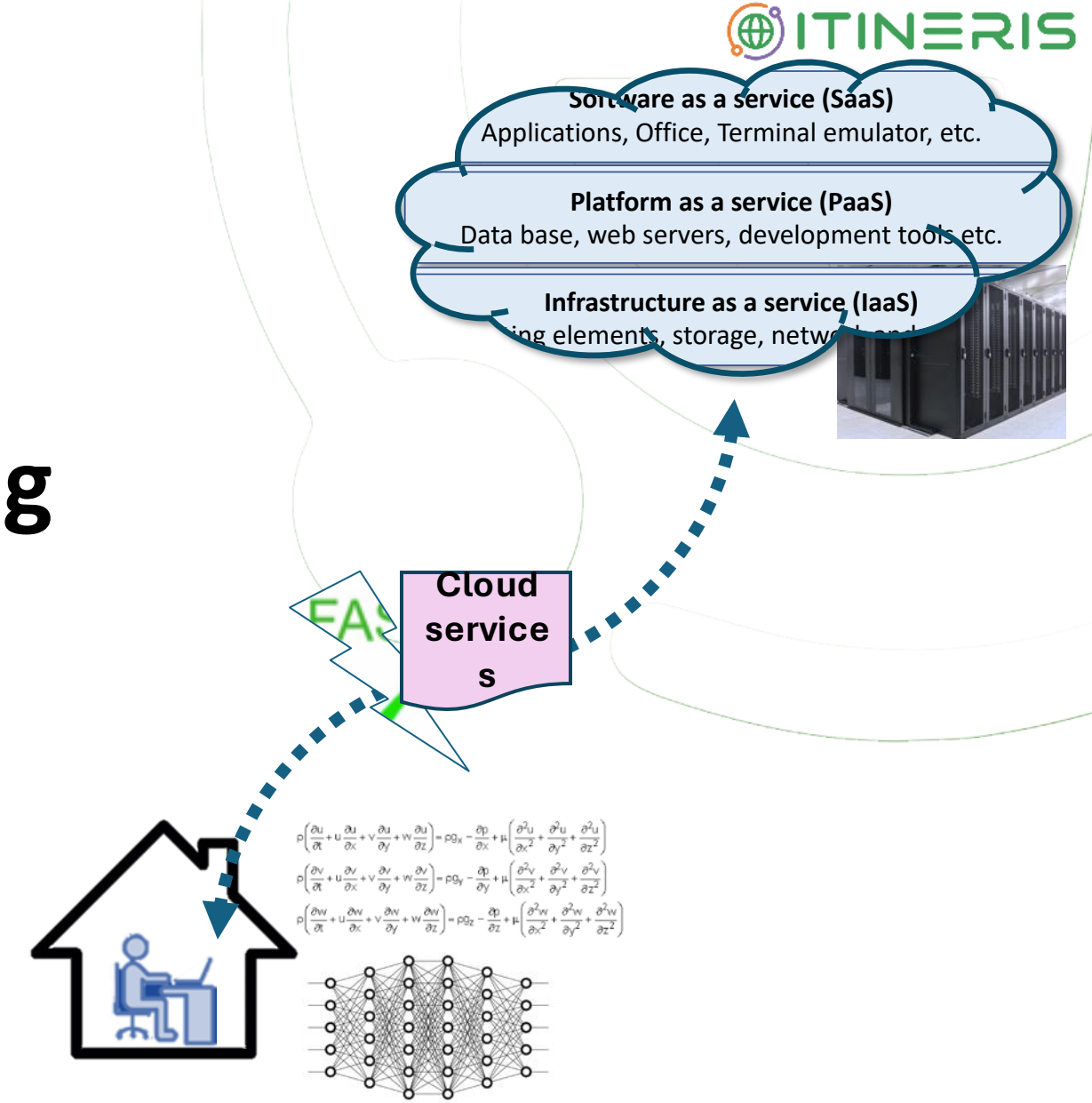
What is Cloud?

- Services of resources
- **Access via Internet**
  - Using internet
  - Access from anywhere
  - Web portal



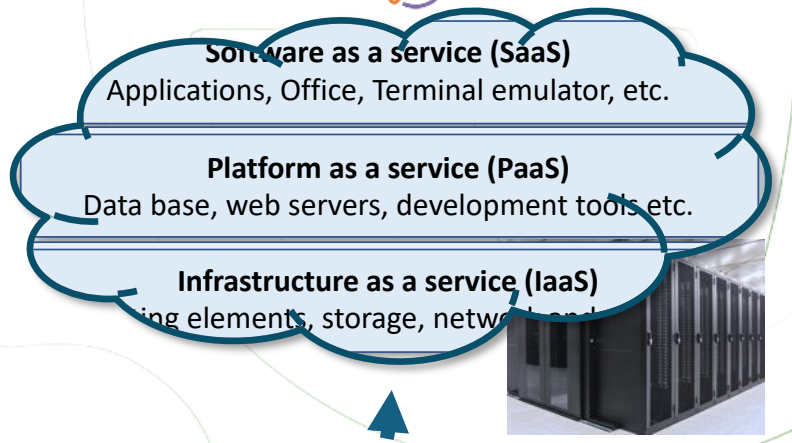
What is Cloud?

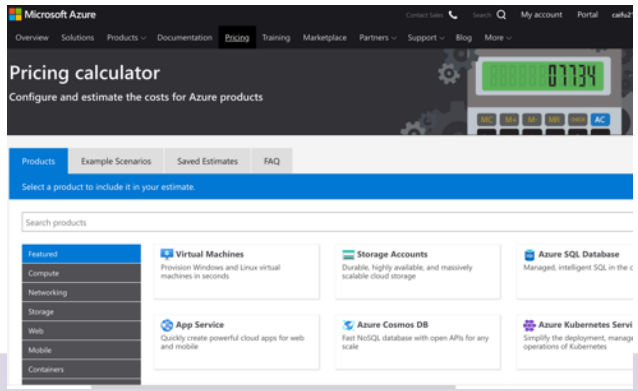
- Services of resources
- Access via Internet
- **On demand provisioning**



What is Cloud?

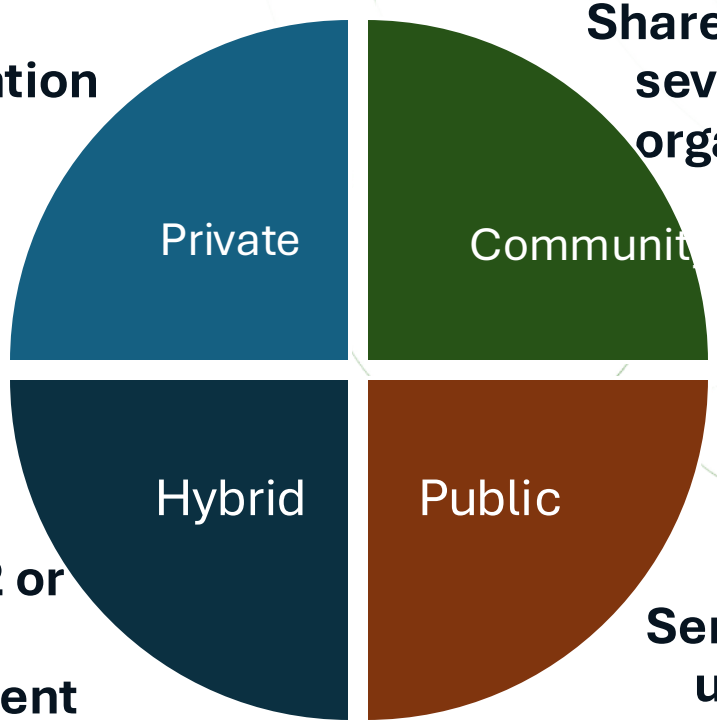
- Services of resources
- Access via Internet
- On demand provisioning
- **Flexible price model**
  - Pay per use
  - Pay as you go
  - Advanced reserved
  - Subscription based





Cloud types

- Private cloud
- Public cloud
- Hybrid cloud
- Community cloud



Single organization

Private used data center, e.g. for single institute

Shared by several organizations

ExoGENI

EGI

...

Mix of 2 or more different types.

See more examples from RIs

Serve public use

Google Cloud Platform

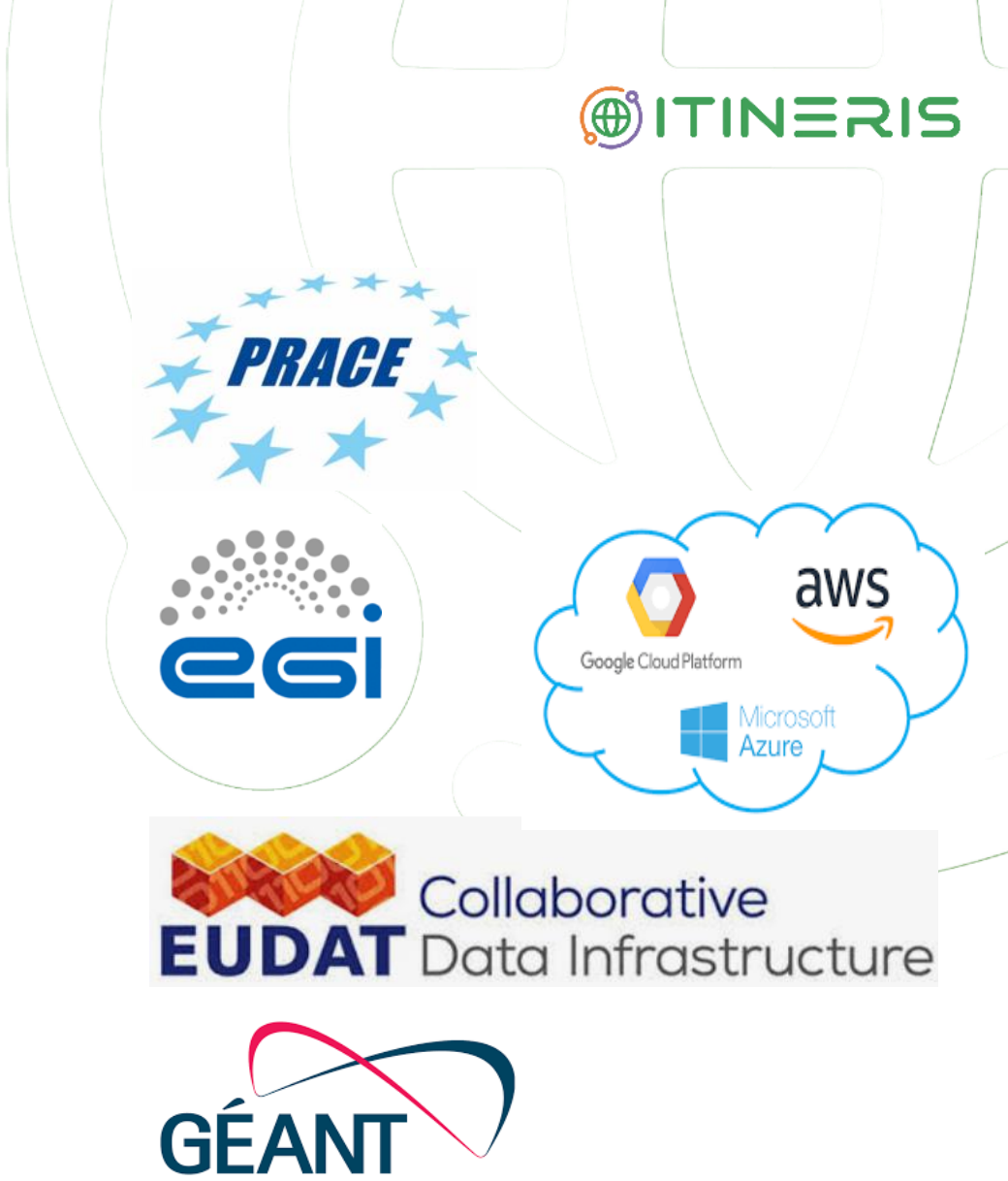
aws

Microsoft Azure

...

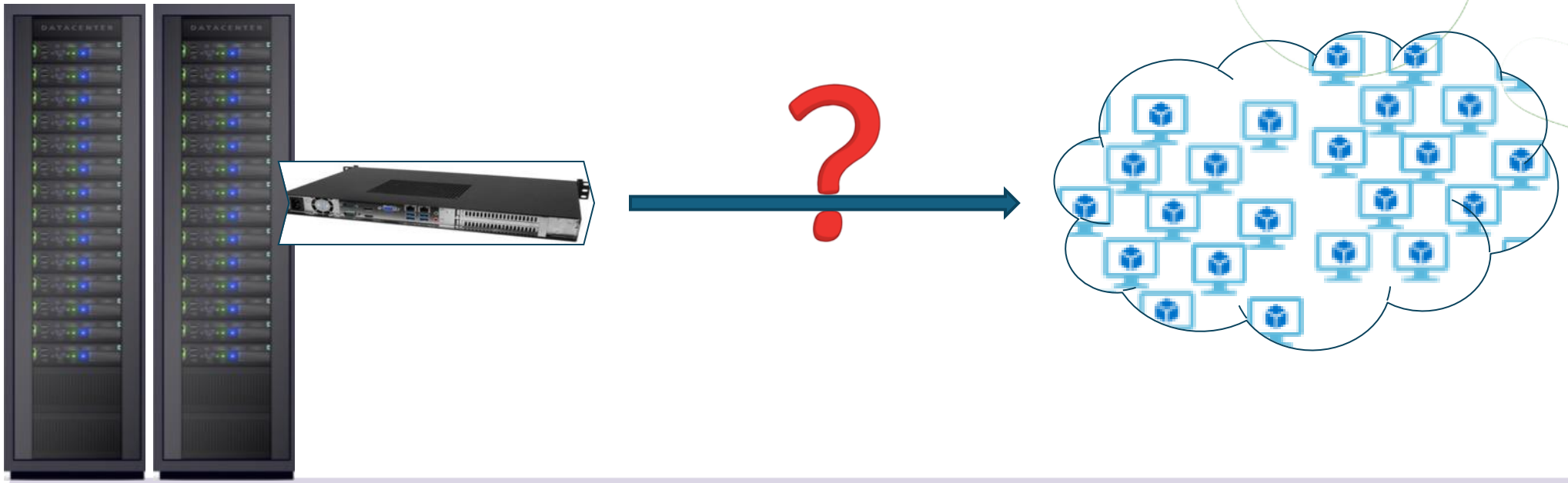
Cloud and e-Infrastructure

- 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



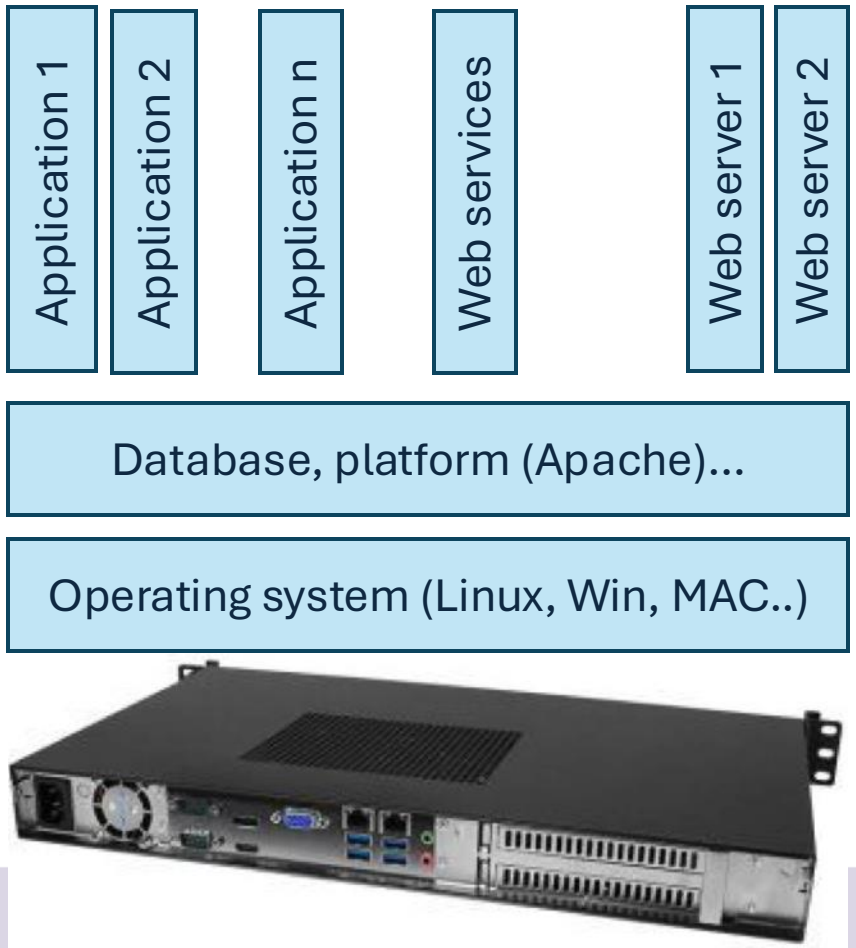
How does cloud work?

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



Virtualization

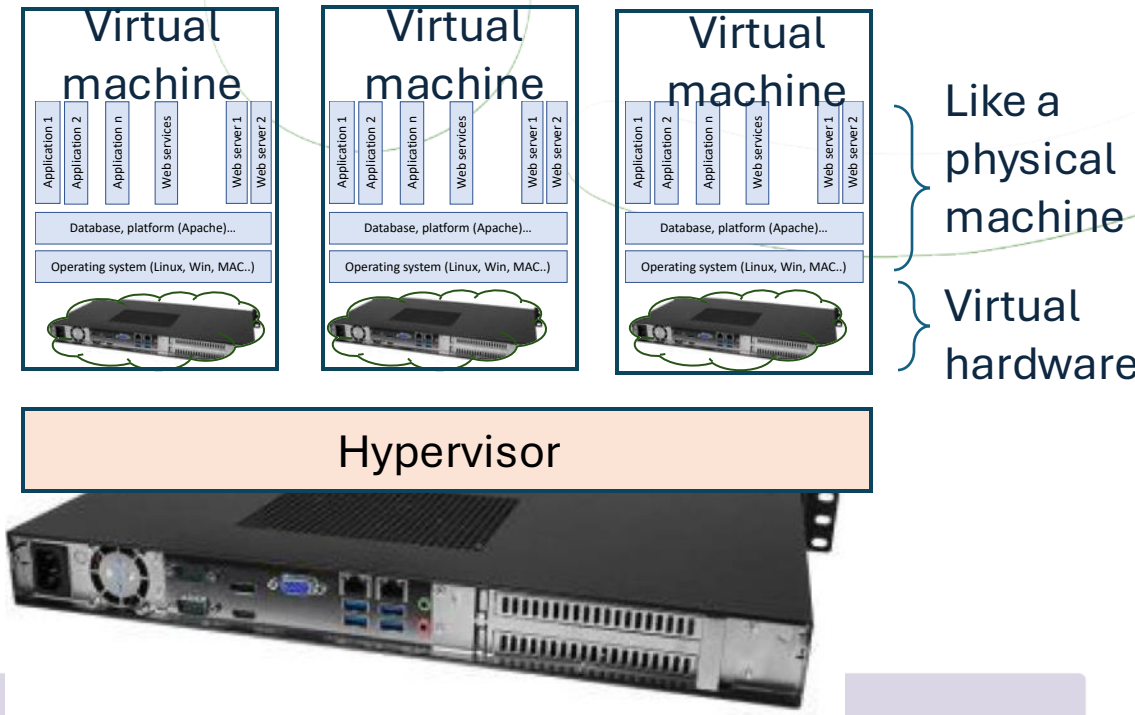
- Technology 1: virtualization
- (Virtual Machine and Hypervisor)



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):

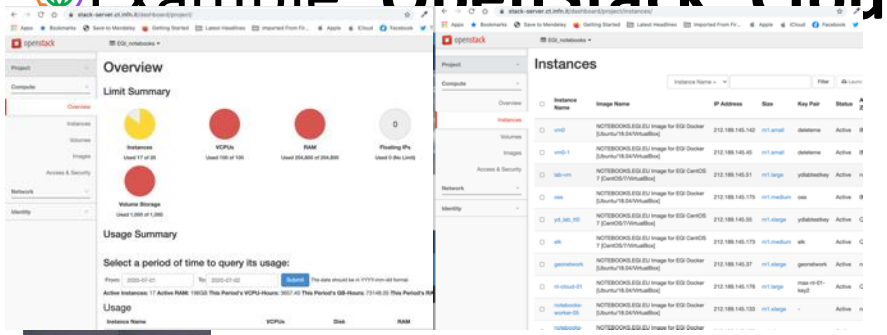
- A software can emulate the behavior of a real computer
- Contains hardware abstraction, OS kernel, library, file systems and etc. The file representation is called VM

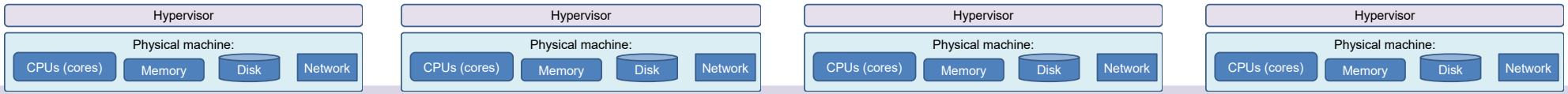
Note:

- ✓ 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)**



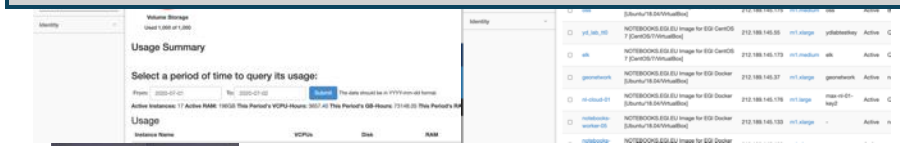


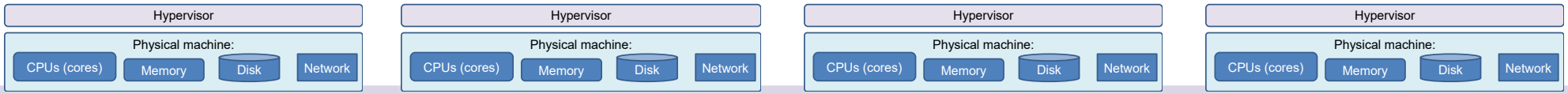
Cont.

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

Note:

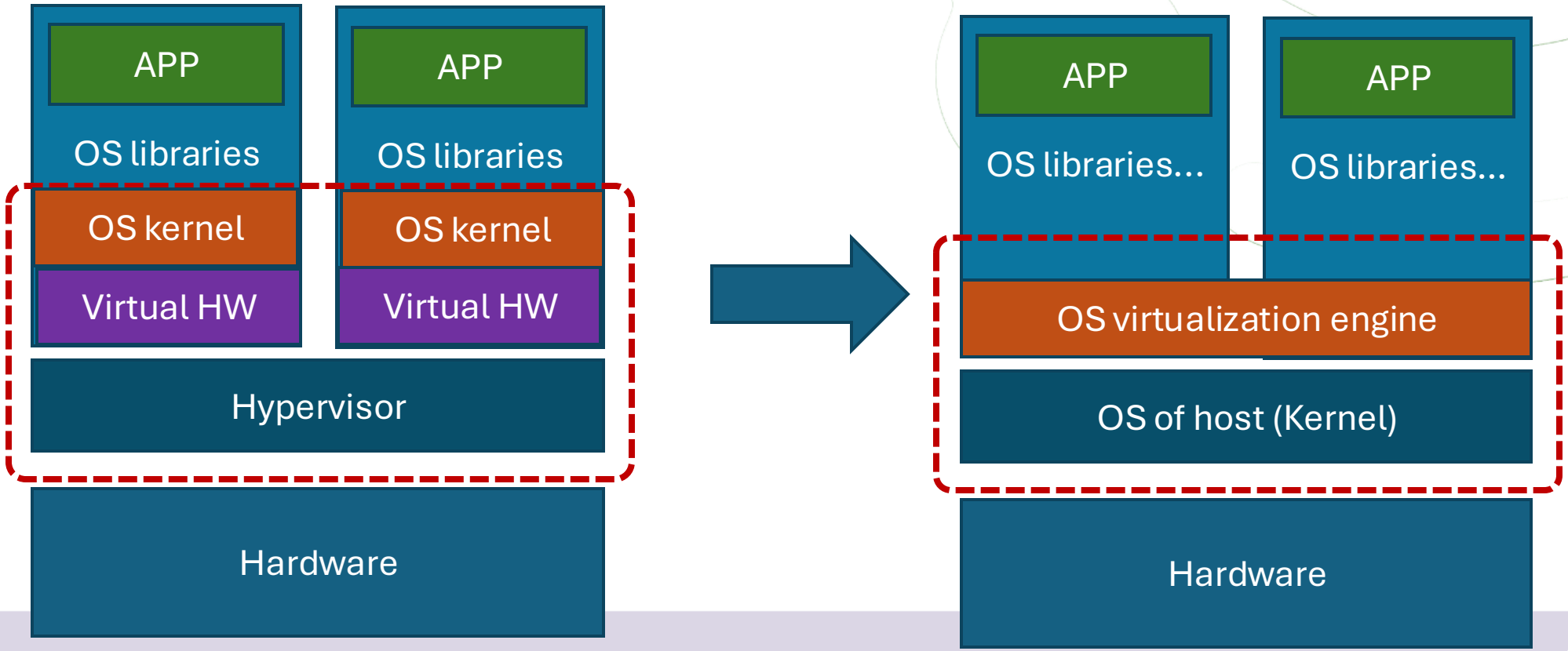
- ✓ An Orchestration system is interacting with the hypervisor, but independent from the hypervisor
- ✓ An Orchestration system provides interface for both users and administrators



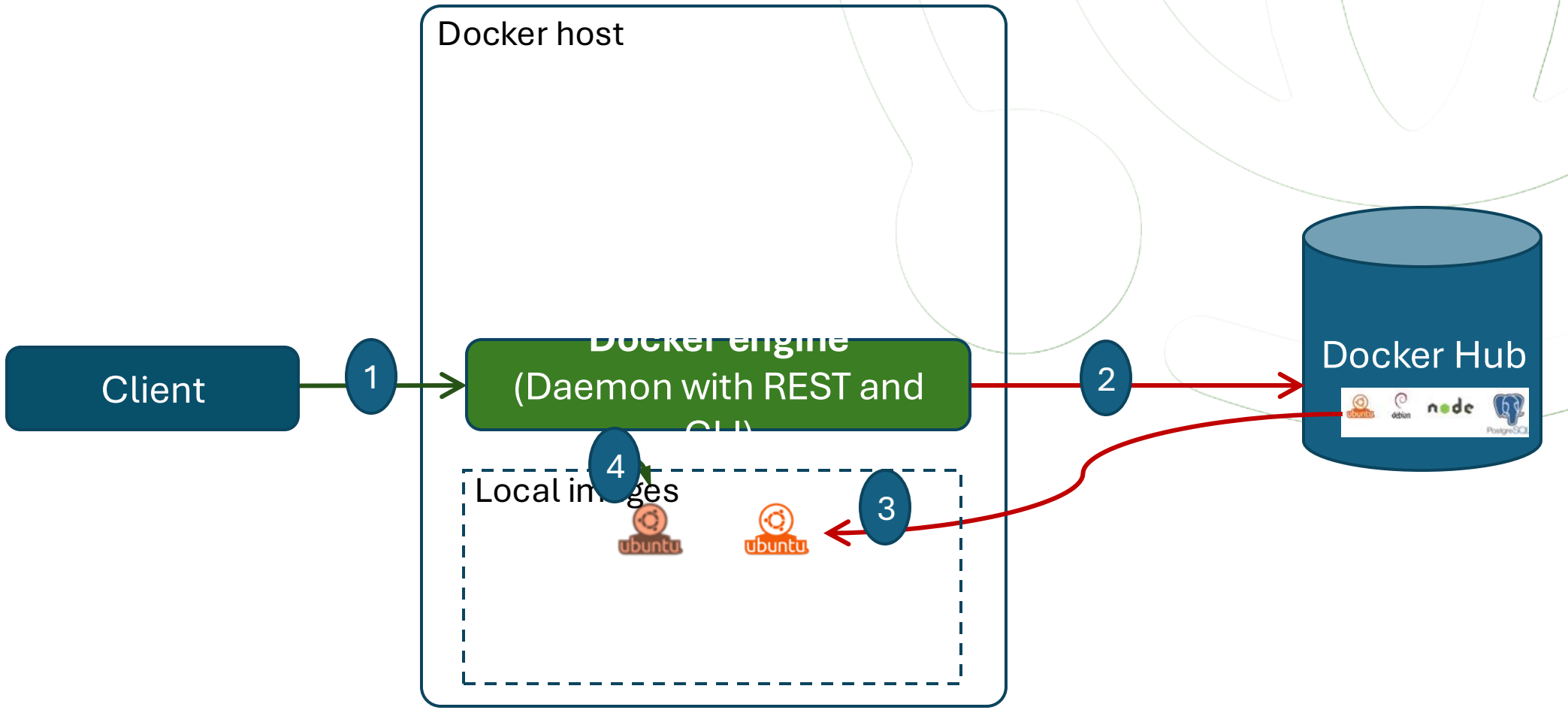


Operating system level virtualization

- Reduce full virtual hardware Shared kernel
- From full virtualization to container

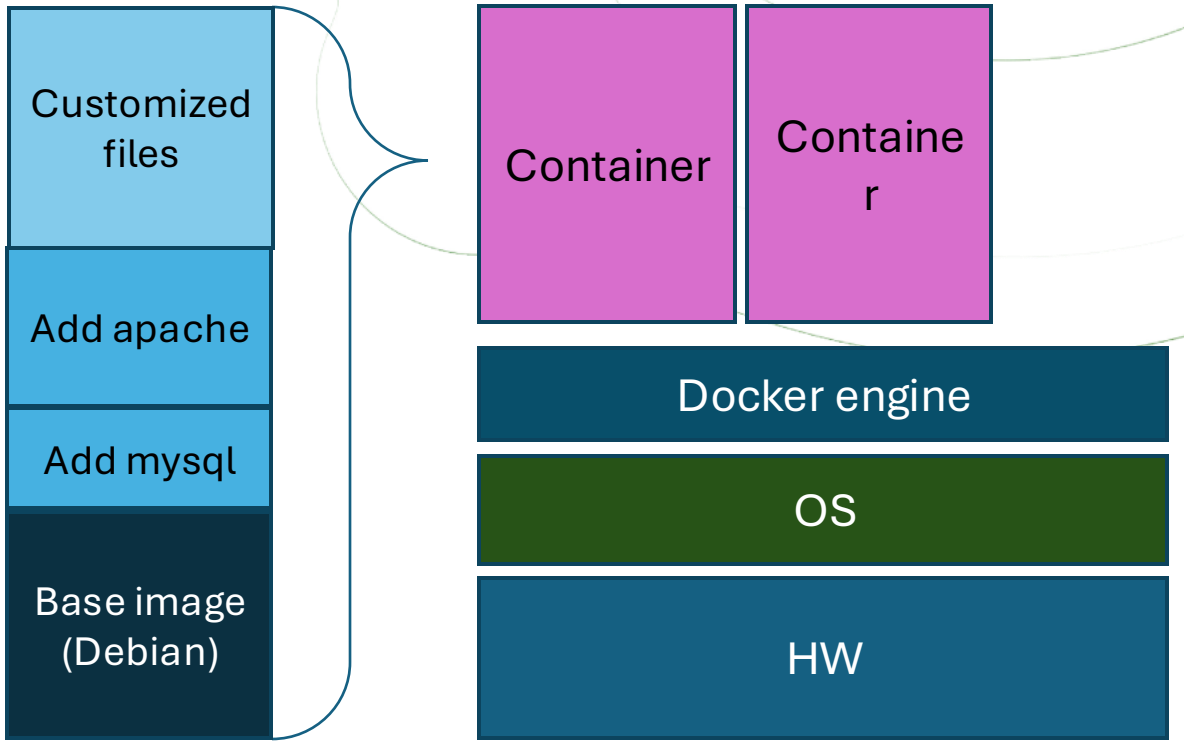


Docker: from image to container
PULL, **BUILD**

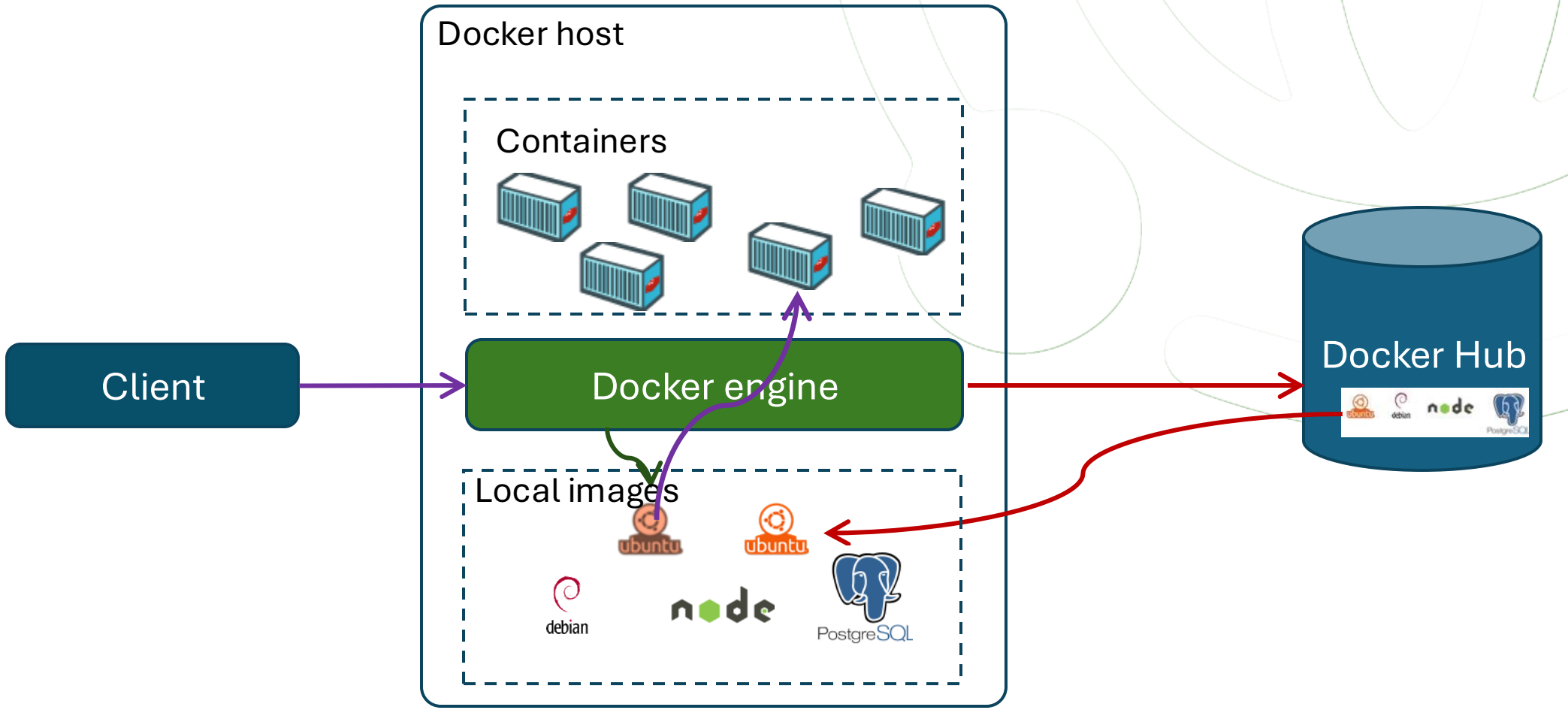


Docker image

- 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

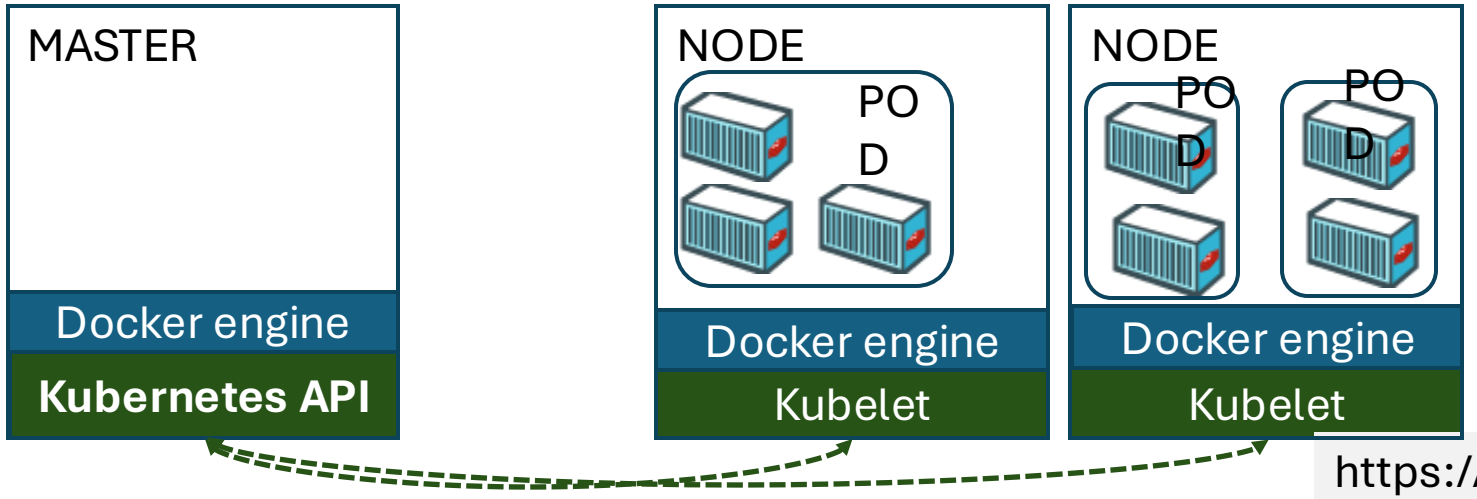


Docker: from image to container
Multiple containers



Kubernetes cluster, POD

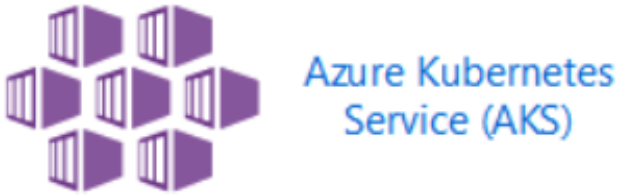
- Smallest unit in Kubernetes, A structural abstraction of a group Containers
  - Some Containers are dependent, and need to be deployed in a single host, or work together. Share IP address or port space.
  - Can also be on container per POD
- Containers in a POD share storage/network



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

Kubernetes utilization

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

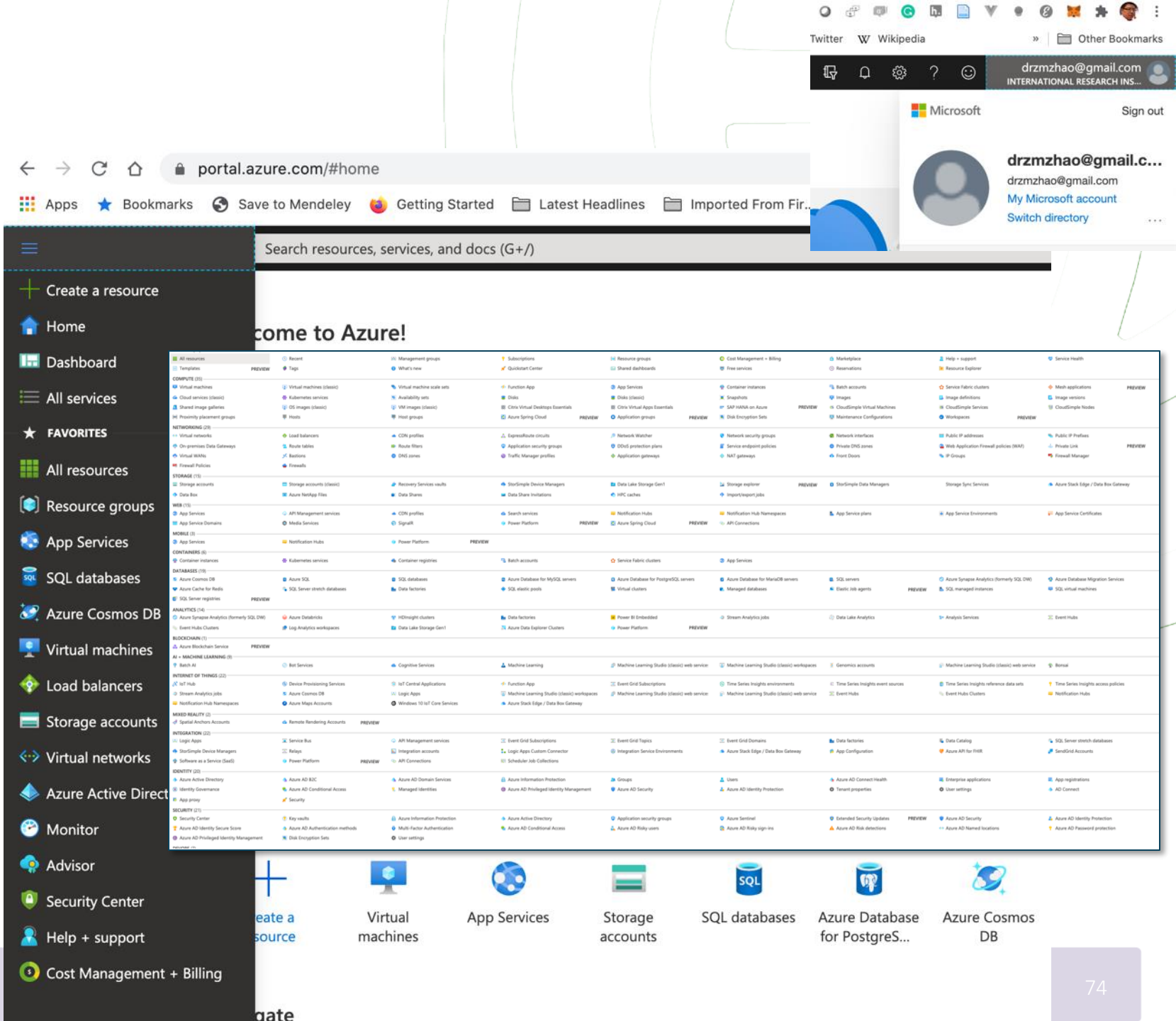


What do we have to consider when choosing cloud service?

- Which provider?
- Which data center?
- What Cloud services?
- What capacity?
- What budget?
- …

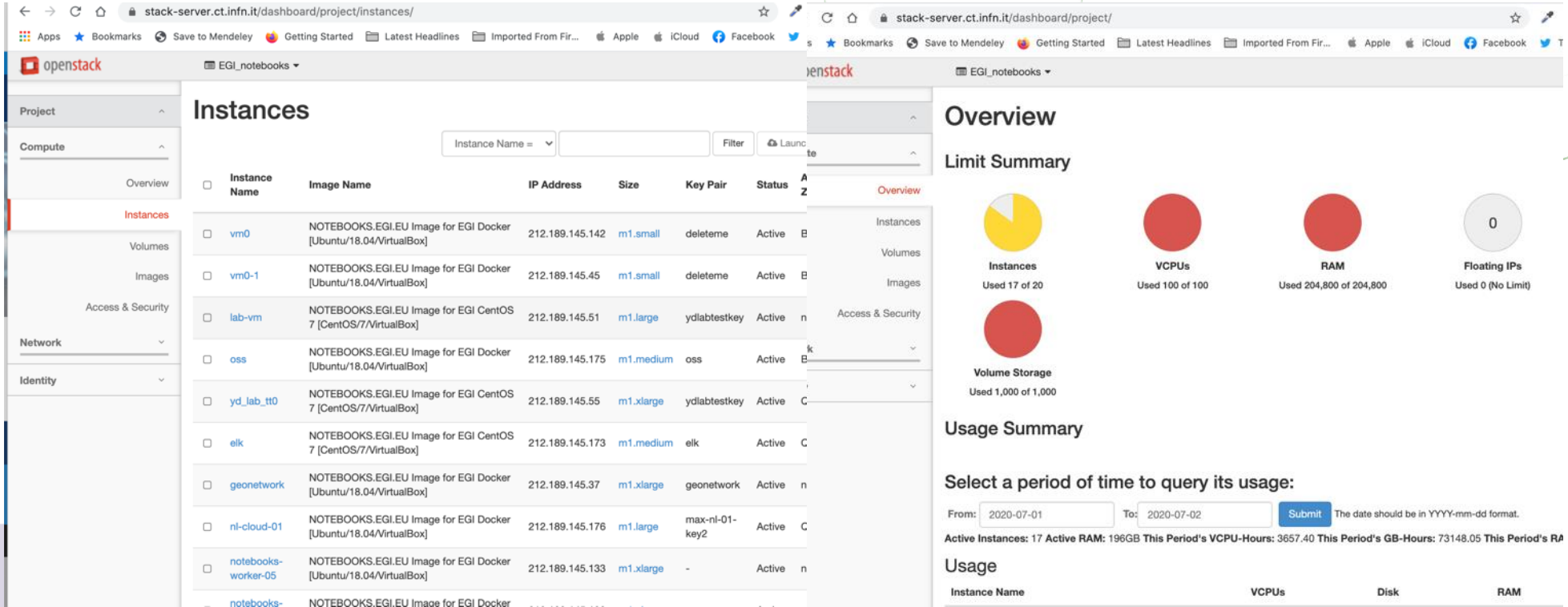
Choose cloud service

- The Cloud Portal (e.g. Azure)
- An account
- Service catalogue
- Order the services and ask for on demand provision



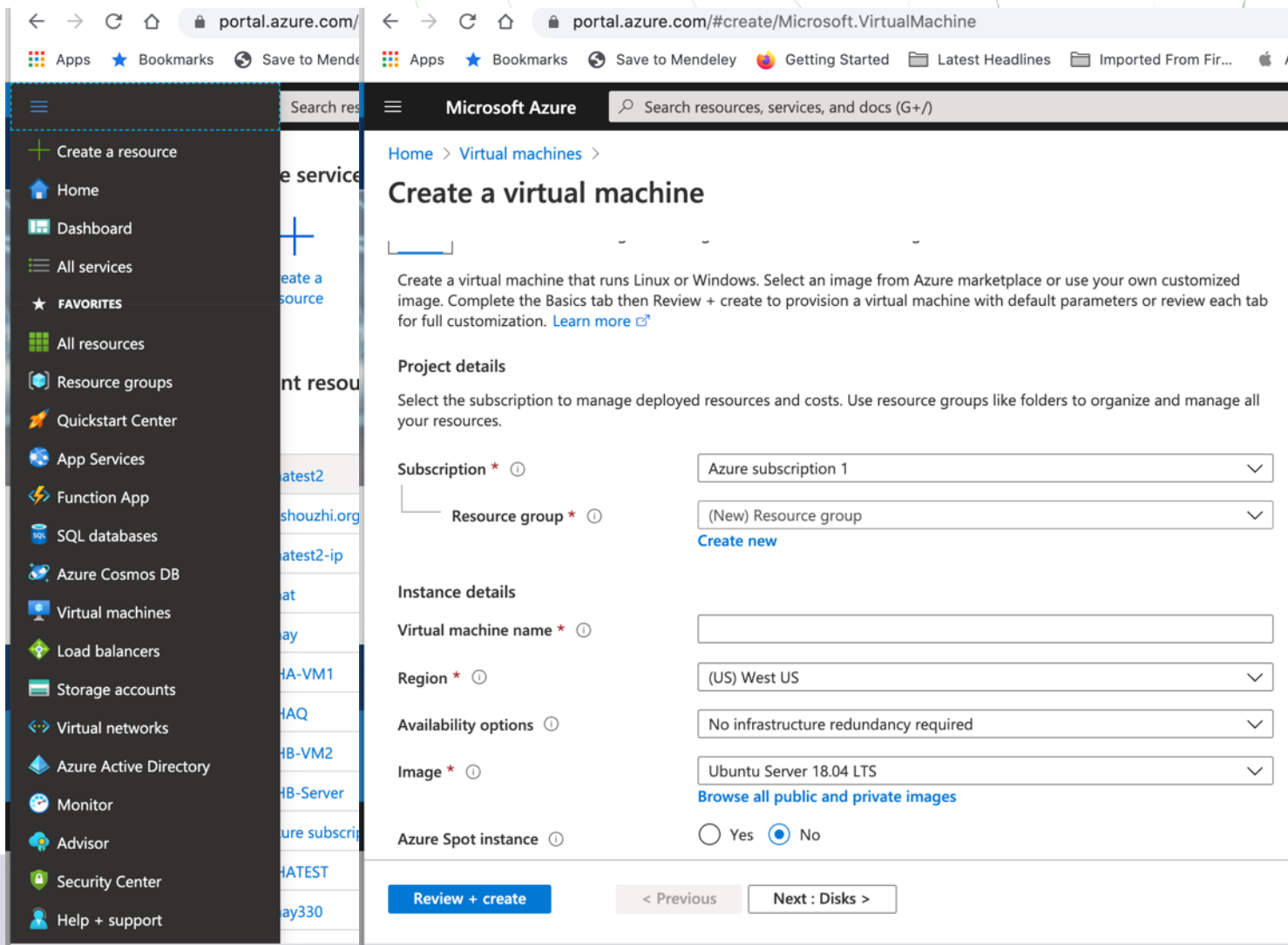
Select cloud services

- Dashboard of the available resources
- Instances
- vCPU
- RAM



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



Discussion

- What is difference between Cloud and a normal computer?

5. Running applications in Cloud

Discussion: How to run your application on cloud?

79

Options for runing 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

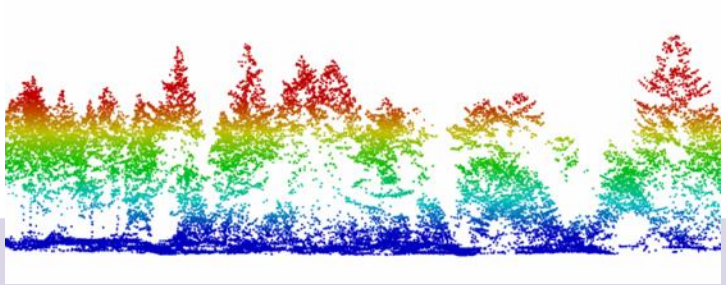
Types of scientific application

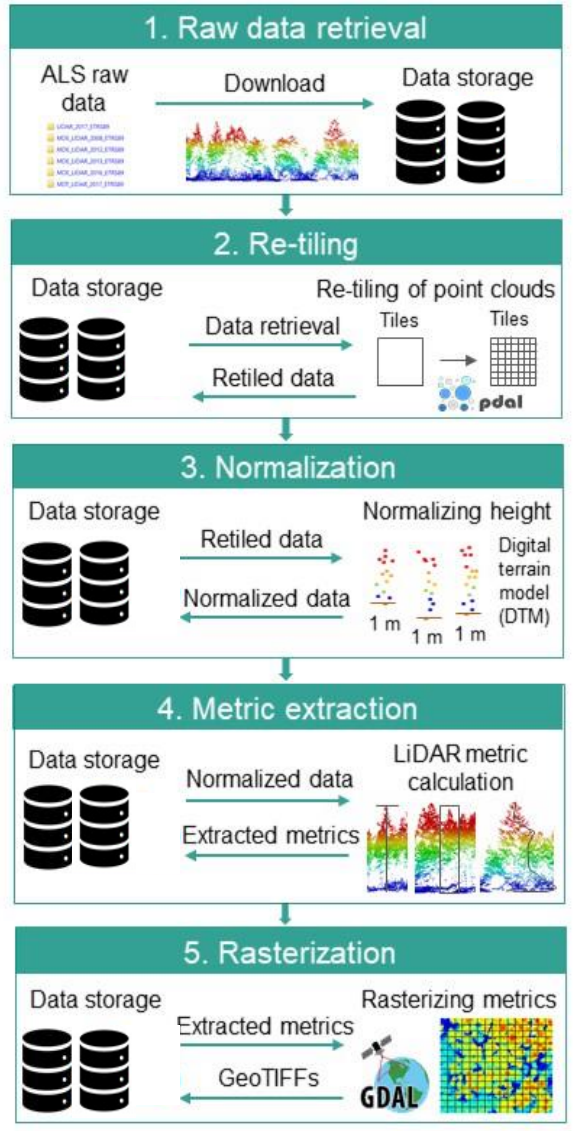
- Modelling and simulation,
- big data analytics,
- machine learning,
- sensor data processing,
- data base
- Workflow of different tasks
- …

An example

- A use case from LifeWatch: processing high-resolution Light Detection and Rar measurements
- A researcher developed basic analysis code in python: point cloud processing f 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)



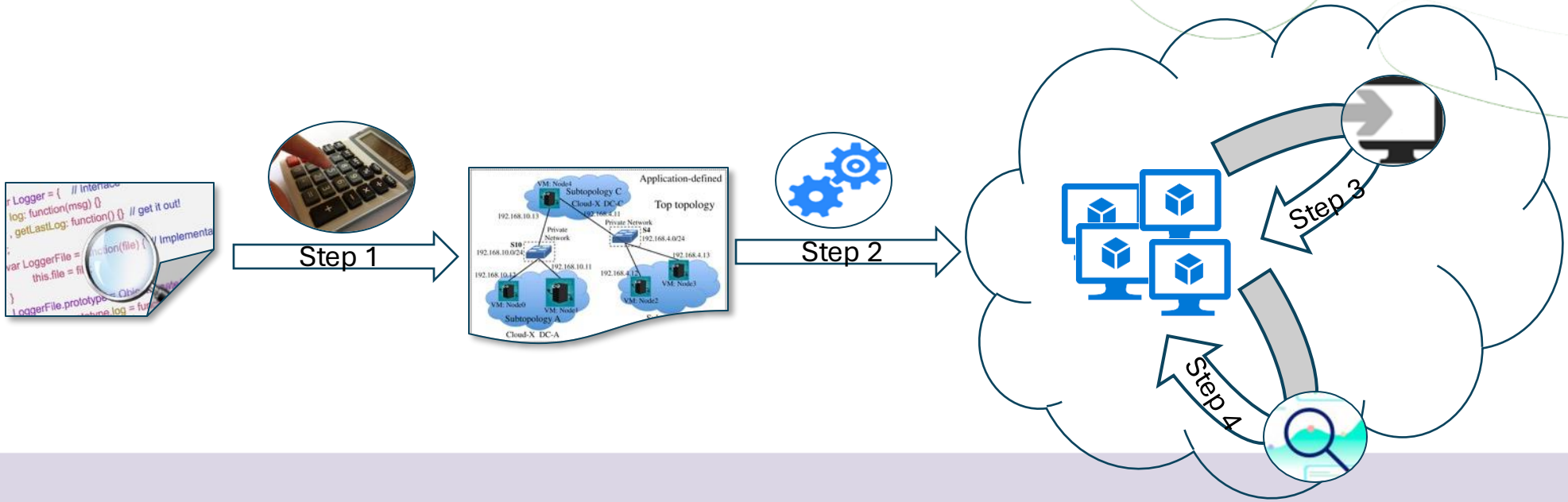


Why use cloud?

- **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 instace
  - simulating high resolution models or training high quality deep neural networks (e.g., combing species distribution or climate information)
- For **combing new features**, or **models**

Option 1: Using Infrastructure services (Virtual machines)

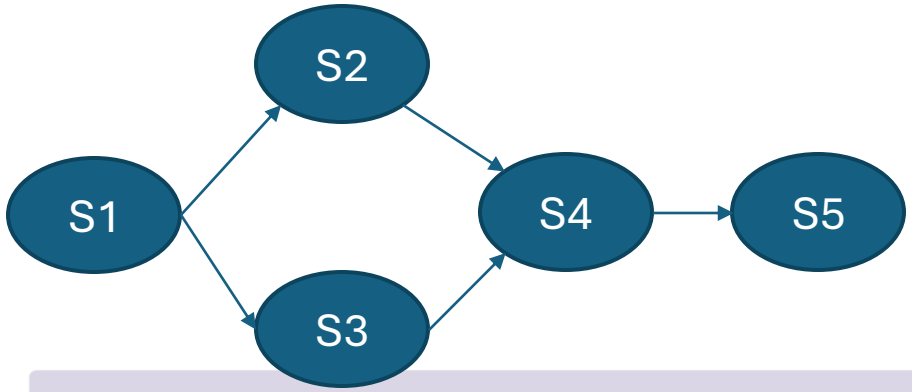
1. Plan Virtual infrastructure (what capacity)
2. Provision Virtual infrastructure (you ask the provider to do it for you)
3. Deploy the Platform and software (you will do it)
4. Execute and monitor the Application (you will do it)



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 characterstics (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 manfully
  - 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-contain, 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?

```
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_DB_NAME:
volumes:
    db_data: {}
```

How does a docker work?

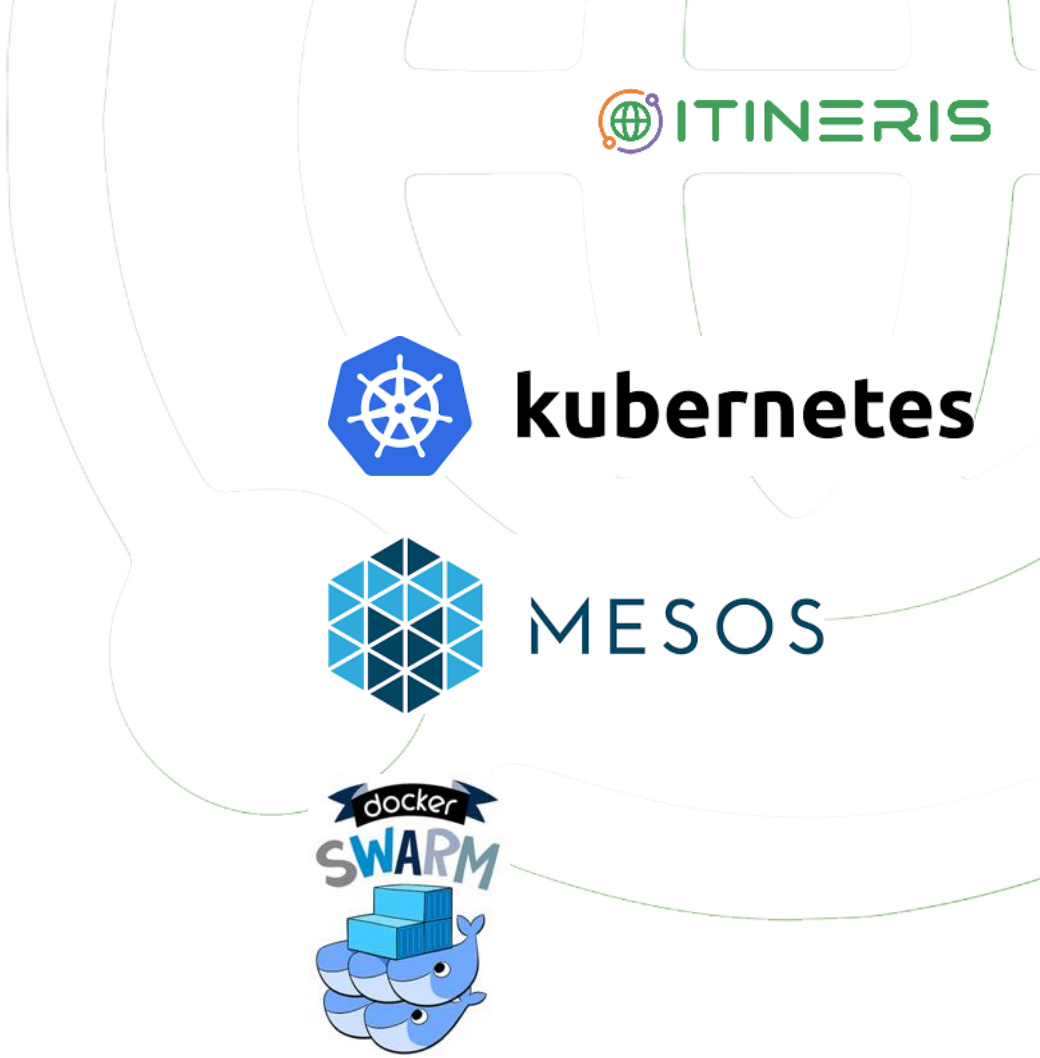
```
version: '3.3'

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_DB_NAME:
volumes:
    db_data: {}
```

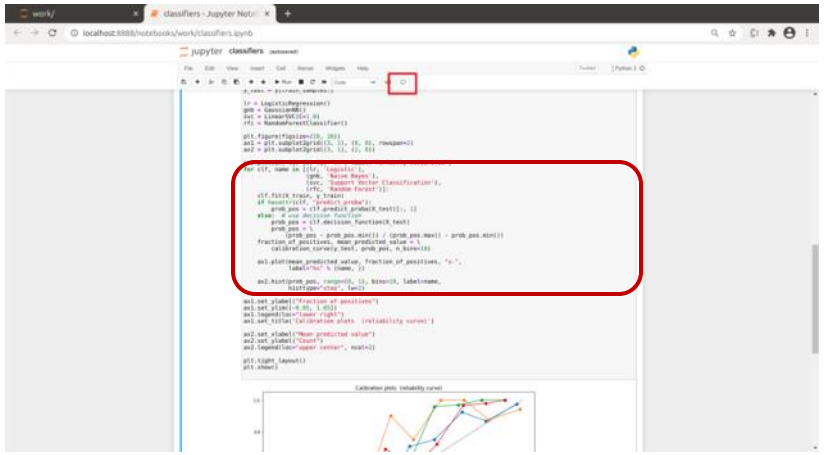
Docker **orchestration** over a cluster

- SWARM: using compose file
- MESOS: using marathon
- Kubernets: google

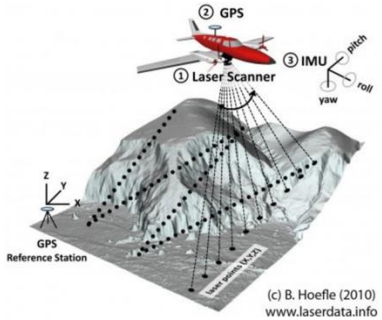


NaaVRE approach

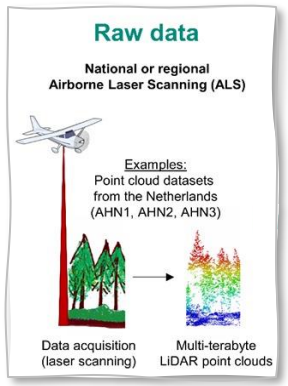
- Scientific code is often developed in Jupyter
- Can we containerize the code, or a specific segment (Cell) as a RESTful service and encapsulate it as a container?

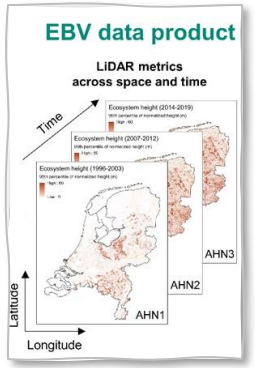


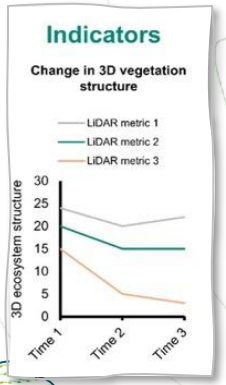
Workflow

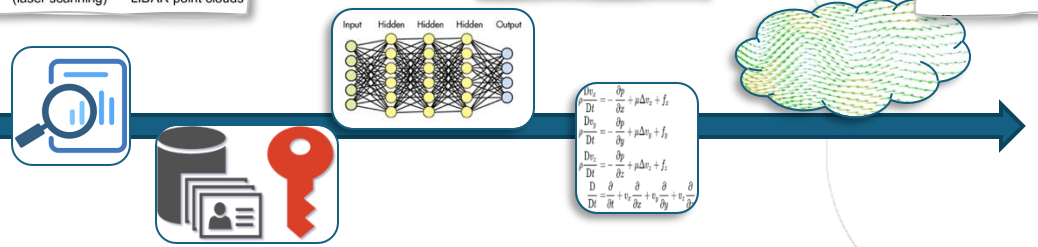


Data









Research &
innovations

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