

DBW – Databases and Web development

Josep Ll. Gelpí. gelpi@ub.edu



Aims

- Review technologies to handle bioinformatics data:
 - Computer communication, design of web applications, basic database design and optimization.
- This is NOT a programming course, it is about designing and building applications in an heterogeneous scenario
- The final objective is to build a **fully operative application** using the appropriate combination of the techniques reviewed.

Bioinformatics & Internet

- Bioinformatics Tools and data should be available through web
- Ex. Nucleic Acid Research reviews:
 - [Database Issue](#) (January)
 - [Web Server Issue](#) (July)



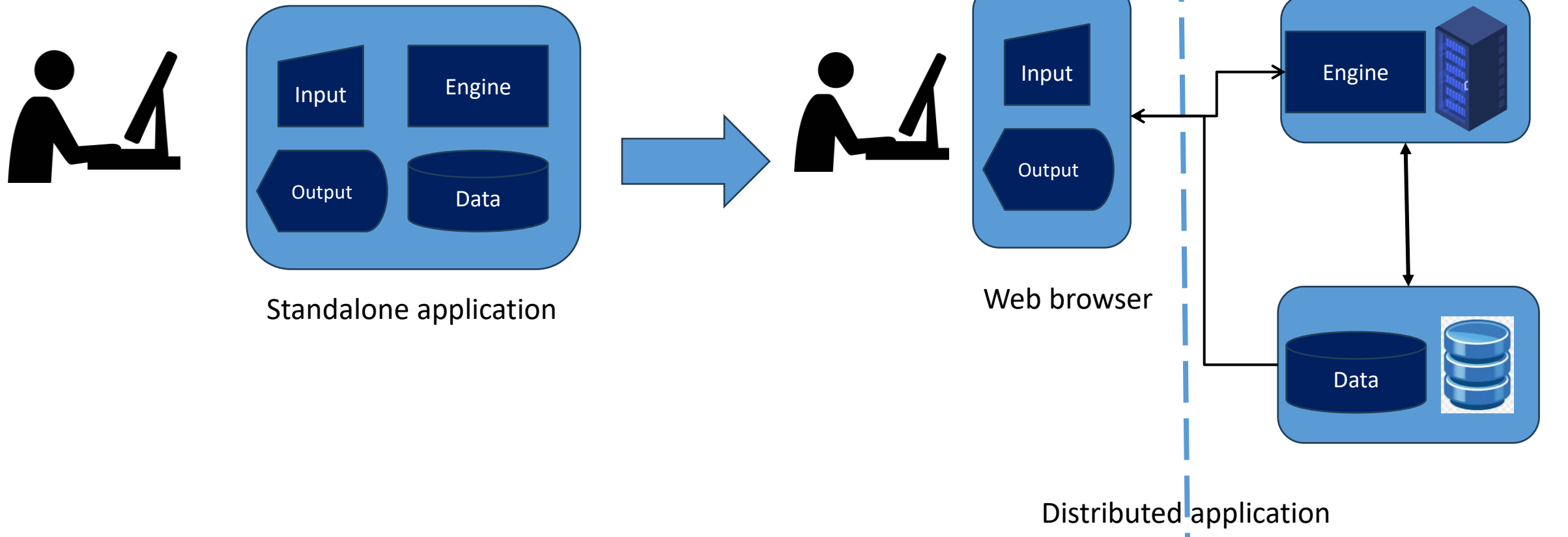
NAR Database issue recommendations for authors

- “The pre-submission enquiry must present a working **web accessible** database “
- “The quality, quantity and originality of data as well as the **quality of the web interface** are the most important. Good data with a poor interface or vice versa are never sufficient for consideration. “
- “**Do get a domain name for your website**. URLs to specific IP addresses/ports are unlikely to stand the test of time.”
- (...)

Nucleic Acids Research, Volume 35, Issue suppl_1, 1 January 2007,
Pages D1–D2

https://academic.oup.com/nar/article/35/suppl_1/D1/1088333

Architecture shift to distributed (web) applications



Web applications by access type

- **Web interfaces**

- Provide a user friendly interface (web based) to “human” users
 - Users known how to use the interface
 - There is no need to install software
 - Single operations (no large scale)

- **Web services & APIs**

- Provide a programmatic interface (using Web protocols)
- Intended to interact with software, not with humans
 - Well-defined data formats required.
 - Adequate for large scale operations

- Modern applications will normally offer both
 - Web frontend is normally just another client of the API's

- **Application styles**

- Access to data
 - Friendly interface to data repositories (aka Data Portals)
- Web Interfaces to stand-alone software
 - Collect input parameters, run, and redirect output
- Workbenches (e.g. Galaxy)
- On-purpose applications & DBs

Data Portals



BLAST Align Peptide search ID mapping SPARQL

Release 2022_05 | Statistics    Help

Find your protein

UniProtKB ▾

Advanced | List Search

Examples: Insulin, APP, Human, P05067, organism_id:9606

UniProt is the world's leading high-quality, comprehensive and freely accessible resource of protein sequence and functional information. [Cite UniProt](#)

Feedback 

Help

Proteins
UniProt Knowledgebase

Species
Proteomes

Protein Clusters
UniRef

Sequence Archive
UniParc

Project Data Portals


Welcome | ICGC Data Portal

https://dcc.icgc.org

Buscar

Microsoft Office Home Documentos de Google Hojas de Cálculo de G... EndNote Aul@-ESCI: Entrar al si... Calendario y horarios Campus Virtual de la ... MICINN - Sede Electrón...

Login

 **ICGC** Data Portal

Cancer Projects

Advanced Search

Data Analysis

DCC Data Releases

Data Repositories

Q e.g. BRAF, KRAS G12D, DO35100, MU7870, FI998, apoptosis, Cancer Gene Census, imatinib, GO:0016049

About Us

The [ICGC](#) Data Portal provides tools for visualizing, querying and downloading the data released quarterly by the consortium's member projects.


To access ICGC controlled tier data, please read these [instructions](#).

New features will be regularly added by the [DCC](#) development team. [Feedback is welcome](#).

Data Release 26

Dec 7th, 2017

Donor Distribution by Primary Site




Cancer projects	76
Cancer primary sites	21
Donors with molecular data in DCC	17,440
Total Donors	20,383

Tutorial

EXAMPLE QUERIES

1. BRAF missense mutations in colorectal cancer
2. Most frequently mutated genes by high impact mutations in stage III malignant lymphoma
3. Brain cancer donors with frameshift mutations and having methylation data available

**PCAWG**
PanCancer Analysis
OF WHOLE GENOMES

The PanCancer Analysis of Whole Genomes (PCAWG) study is an international collaboration to identify common patterns of mutation in more than 2,800 cancer whole genomes from <https://dcc.icgc.org/icgc-in-the-cloud> Consortium.

ICGC
International
Cancer Genome
Consortium


in the cloud

ICGC data is now available on commercial and [Go to ICGC in the Cloud Home](#)

Or both...

The screenshot shows the PMut web application interface in a browser window. The URL bar shows `mmb.irbbarcelona.org/PMut`. The navigation bar includes links for **PMut**, **Repository**, **Analyze mutations**, **Batch analysis**, **Custom predictor**, **REST API**, **PyMut**, **Help**, and **Login**.

Welcome to the updated version of Pmut (Beta), you can find the old version [here](#).

Predict the pathology of mutations
Enter a list of mutations on any protein or sequence, and find out their pathology score using PMut predictor.
[Analyze mutations »](#)

Do you have lots of mutations? Submit a batch analysis!
If you want to predict lots of mutations on different proteins, you can submit a batch query.
[Batch analysis »](#)

Browse our repository
We have a repository of 725,596,928 variants on 106,407 proteins that have been analyzed and are predicted to be either pathological or neutral.
 [Search](#)
e.g. `BRCA2, 2vgb, ENSG00000133110, Q04917`

Train your own predictor
Train a specific predictor using your own annotated variants and get more precise predictions for your research.
[Train custom predictor »](#)


Server status

- 0 queued jobs
- 0 running jobs
- ✓ 873 completed jobs
- 725,596,928 variants
- 106,407 proteins
- 17 registered users

Download PyMut to work locally
The [PyMut](#) Python module brings all the PMut functionality to your computer, allowing you to customize and tinker the machine learning process. Check the [PyMut tutorial](#) as an example of how to use it.

Contact
If you have any question or request about this service, contact us at pmut@mmb.irbbarcelona.org

Web interfaces to bioinformatics applications

 EMBL-EBI


Services


Research

Training

Industry

About us



EMBL-EBI  Hinxton ▾

Clustal Omega

Input form

Web services

Help & Documentation

Bioinformatics Tools FAQ

 Feedback

Tools > Multiple Sequence Alignment > Clustal Omega

Multiple Sequence Alignment

Clustal Omega is a new multiple sequence alignment program that uses seeded guide trees and HMM profile-profile techniques to generate alignments between **three or more** sequences. For the alignment of two sequences please instead use our [pairwise sequence alignment tools](#).

Important note: This tool can align up to 4000 sequences or a maximum file size of 4 MB.

STEP 1 - Enter your input sequences

Enter or paste a set of

PROTEIN ▾

sequences in any supported format:

Workbenches

Galaxy / ELIXIR-ES

Flujo de Trabajo Visualizar Datos Compartidos Administración Ayuda Usuario

Utilizando 911.8 M

Herramientas

Buscar herramientas

Cargar Datos

Get Data

Send Data

Collection Operations

Text Manipulation

Convert Formats

Filter and Sort

Join, Subtract and Group

Fetch Alignments/Sequences

Statistics

Graph/Display Data

BIOEXCEL BUILDING BLOCKS

Get Data

Haddock

Structure Utils

Setup and Simulation (GROMACS)

Welcome to biobb.usegalaxy.es, the INB's Galaxy server for the BioExcel Building Blocks software library.

bioexcel

bioobb

BioExcel Building Blocks, a software library for interoperable biomolecular simulation workflows

The BioExcel Building Blocks (biobb) software library is a collection of Python wrappers on top of popular biomolecular simulation tools, adapted here to be run on Galaxy. The library offers a layer of interoperability between the wrapped tools, which make them compatible and prepared to be directly interconnected to build complex biomolecular workflows.

BioBB Galaxy tools

BioBB demonstration workflows (including Galaxy)

Additional servers for BioBB's:

BioBB REST API

BioBB Workflows Web portal

BioExcel Building Blocks, a software library for interoperable biomolecular simulation workflows. P. Andrio, A. Hospital, J. Conejero, L. Jordá, M. Del Pino, L. Codo, S. Soiland-Reyes, C. Goblet, D. Lezzi, R. M. Badia, M. Orozco & J. Ll. Gelpi. Scientific Data, 6(1),169 (2019)

ELIXIR

INSTITUTE FOR RESEARCH IN BIOMEDICINE

BSC Barcelona Supercomputing

Historial

buscar conjuntos de datos

RSV 5C69

104 shown, 114 deleted

850.44 MB

218: 5c69Trimer_10ns.tpr

217: mygmx_trjconv_str.gr

216: mygmx_image.trr

215: mygmx_rgyr.xvg

214: mygmx_rms.xvg

213: mygmx_rms.xvg

212: mymdrun.xvg

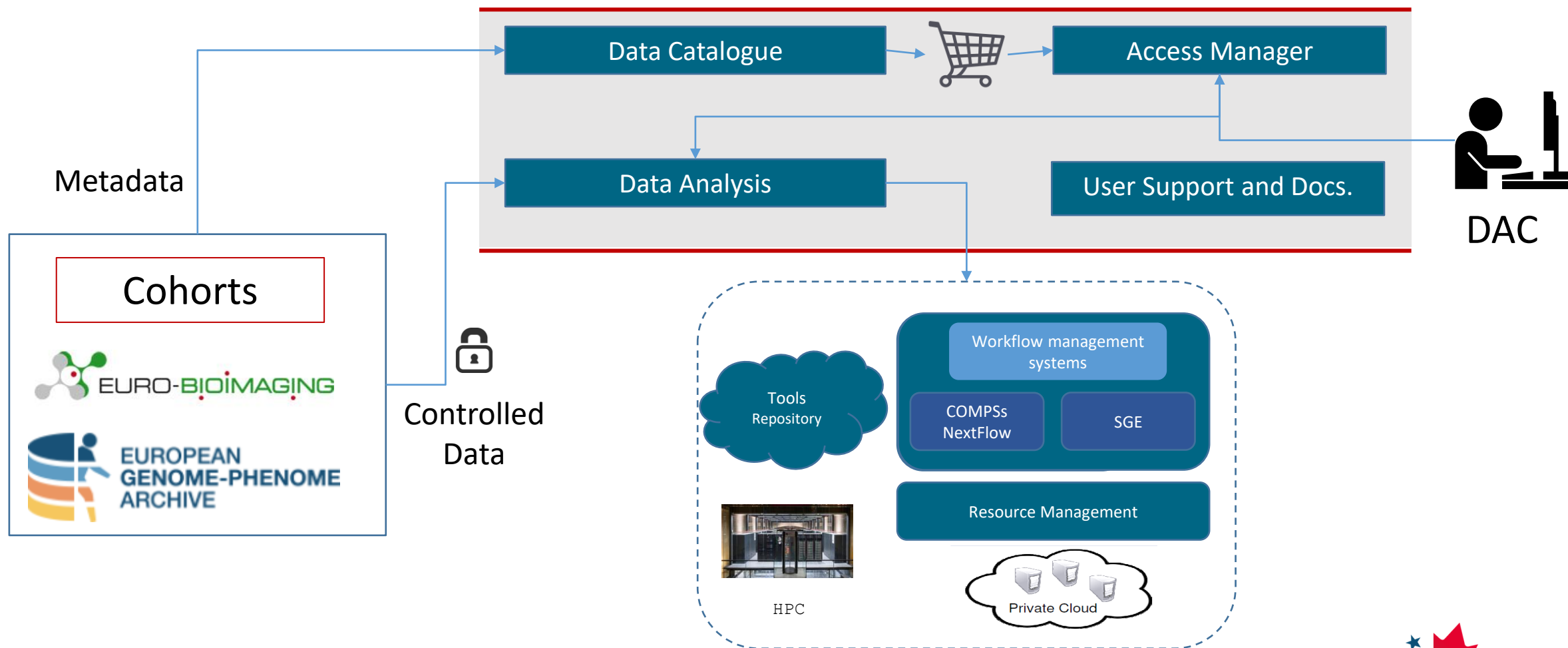
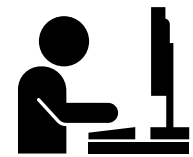
211: mymdrun.cpt

210: mymdrun.xtc

209: mymdrun.log

208: mymdrun.edr

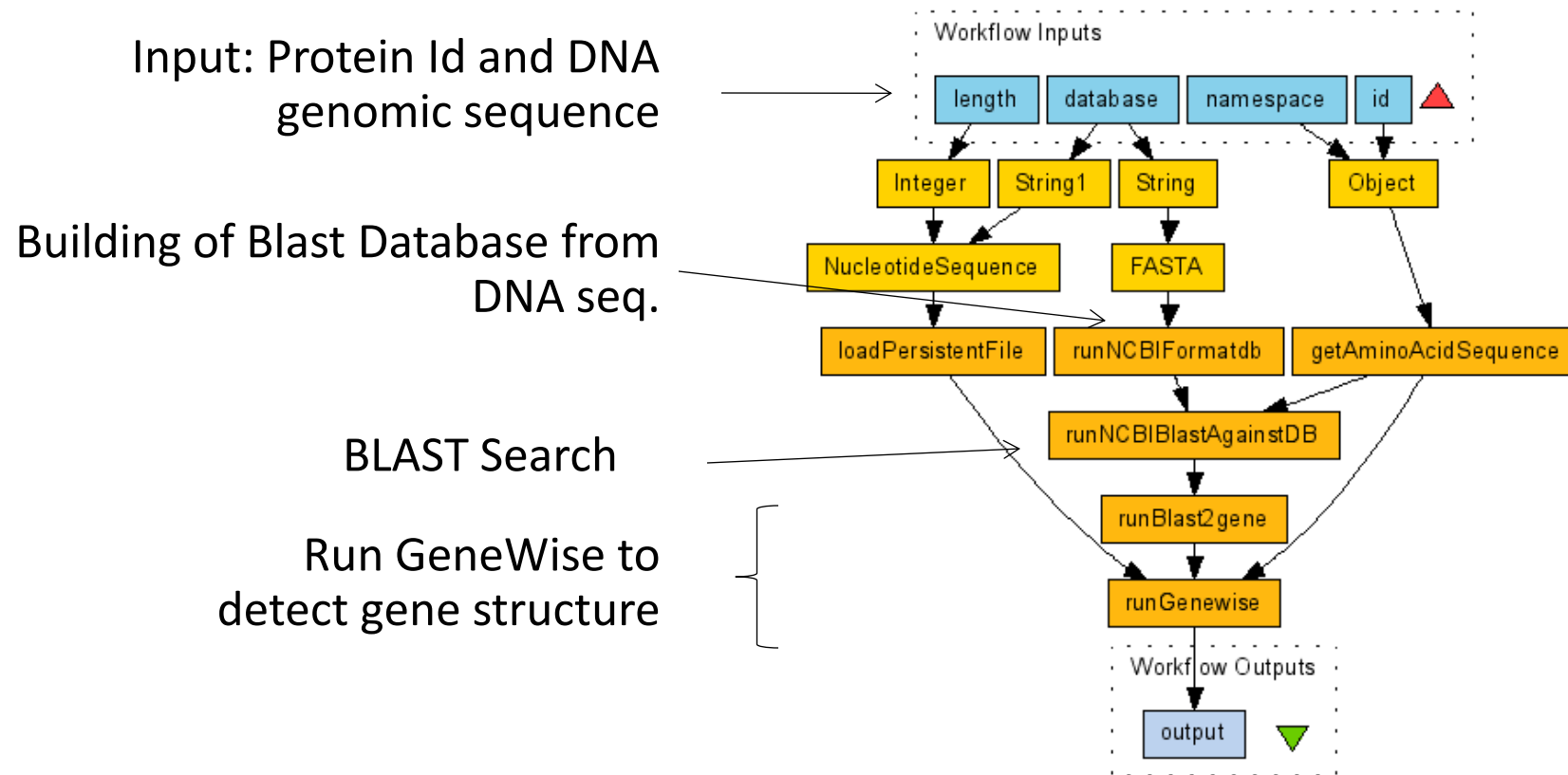
Integrated platforms



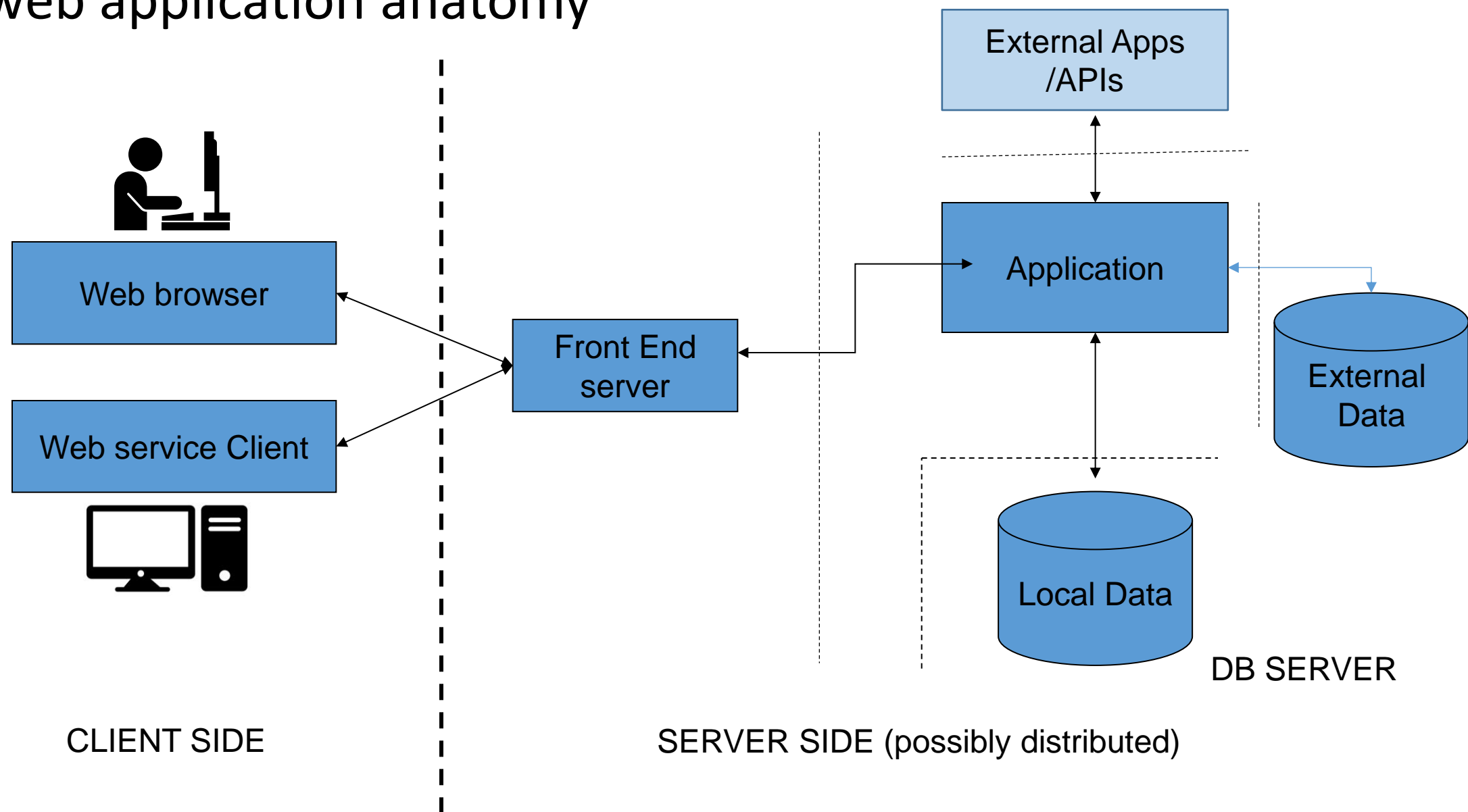
<https://eucanshare.bsc.es>



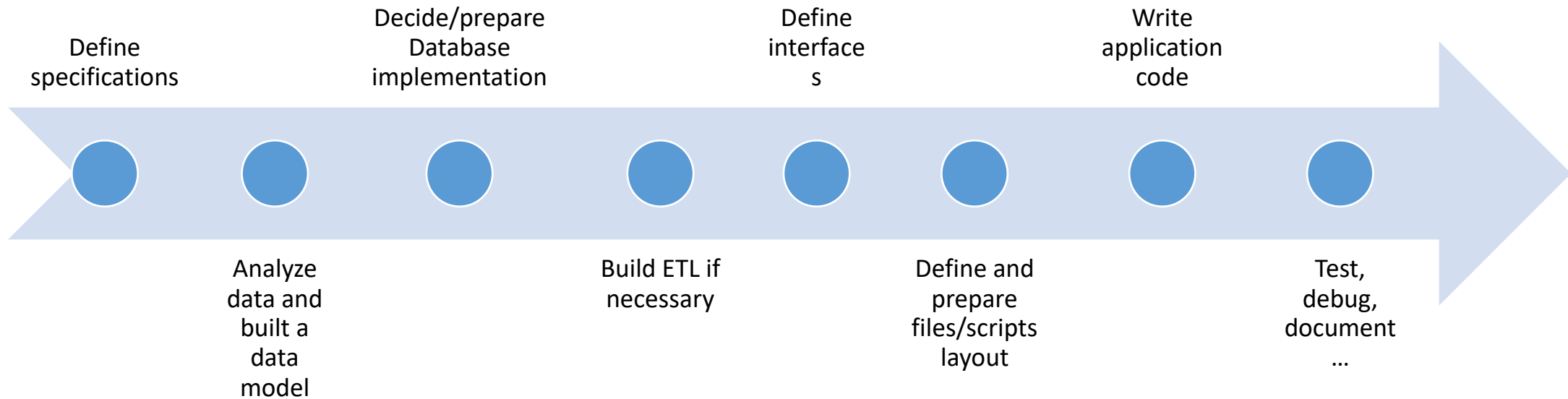
Bioinformatics web-services and workflows



General web application anatomy



Building a (web) application, usual steps





- Web site(s)

- Course materials:

- <https://formacio.bq.ub.edu/>

- Personal sites:

- <https://formacio.bq.ub.edu/~uXXXXXX>

- Server

- SSH Access

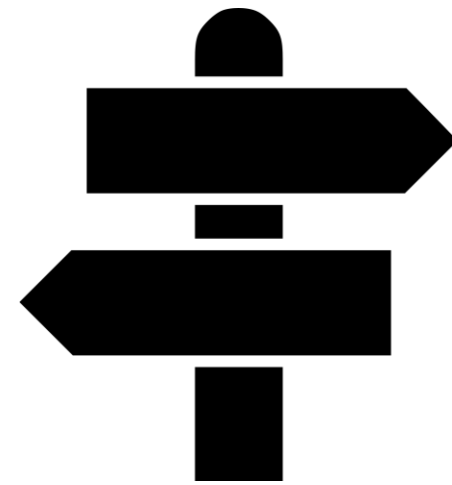
- `ssh formacio.bq.ub.edu -l uXXXXXX`
 - Password `dbw_uXXXXXX`

- SCP

- `scp uXXXXXX@formacio.bq.ub.edu ...`

- MySQL/MongoDB Access

- Localhost only
 - DBs on demand





Software to install

- Ideally Linux
 - Also Windows WLS or Mac
- From Linux distributions
 - A Web server (one of)
 - Apache (with PHP 7.x)
 - Nginx (better for Python apps)
 - MYSQL (or MARIADB) server
 - MYSQL Workbench or phpMyAdmin
- Your preferred software editor
- MongoDB (optional)
 - Install drivers for PHP/python if needed



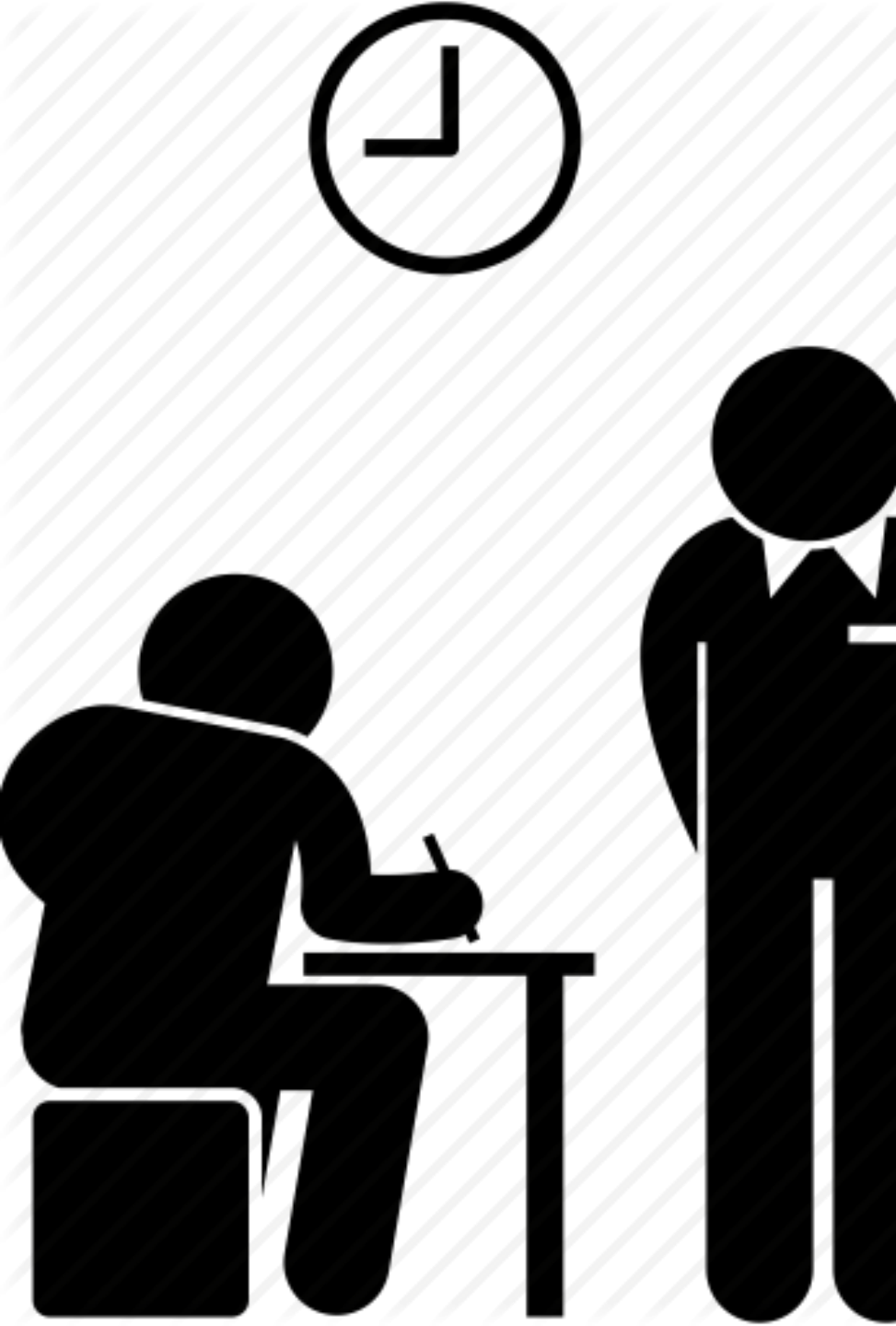
Evaluation

- Personal web site (20%)
- Exercices, in-class projects (20%)
- Web application project (60%)
 - Progress presentations
 - A fully operative web application using a local Database



Evaluation

- Web application project
 - 3-4 people / group
 - Free subject (bioinformatics preferred)
 - **Should include DB management, web interface, users' management** (Mysql or MongoDB)
 - May use **fake data if necessary**
 - Available at the personal web sites of all team members
 - Preferred languages: PHP, Python, ...
 - Source code at github or equivalent



Evaluation

- Web application project
 - Steps (and Deadlines):
 - Initial specification (Presentation 20th Jan)
 - Data analysis & Database design (Presentation 27th Jan)
 - Project prototype Demo (Presentation 3th Feb)
 - Mid development review (meeting around 20th Feb)
 - Final application (End of Term)
- Fully Installed and functional **on course server**
 - PHP projects will use Apache
 - Python projects will use uwsgi/nginx, uvicorn,... (dev servers not acceptable)

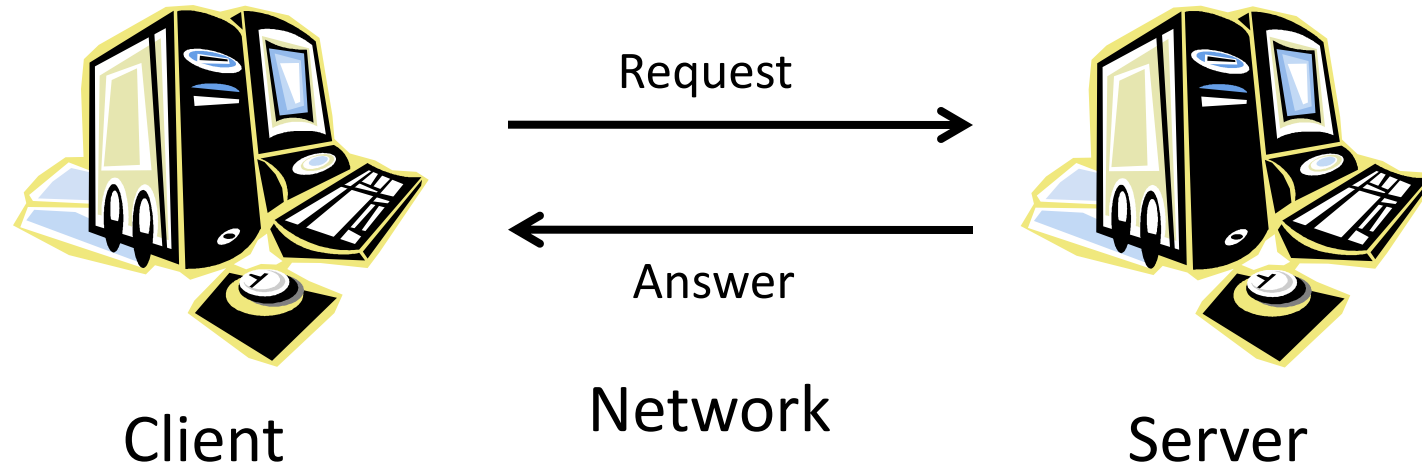
Basic computer communication protocols



Aim & Outline

- Understand the basic components of computer communication protocols
 - Concepts of client and server
 - Addressing servers and data
 - Computer addresses (MAC Address, IP Address, DNS)
 - Ports
 - Resource identification: URL/URI concepts
 - Client/server transactions
 - HTTP protocol

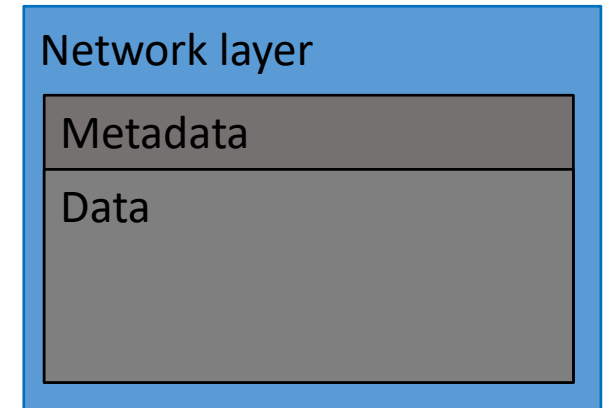
Some definitions ...



- Clients and Servers require logic and physical addresses
- Clients and servers refer both to computers and to software components
- Data transferred requires MetaData (meta-information)
 - Nature of data
 - Request (what to do)
 - Applications involved (email, web, etc.)

How it works: TCP/IP

- Packet switching
 - Packet switching breaks the signal in small fragments
 - Users have the idea of a dedicated line but, in fact, it is not.
- Computers connected to internet should have addresses/ports
 - MAC Address: Address of the physical interface
 - IP Address: Unique address of the computer
 - Unique Host name
 - Ports to point to specific applications



Identifying computers

IP Addresses

- Allow to find destination irrespective of the nature of the network media.
- Each device has a **“unique” IP address**
- IPv4: 32 bits (4 x 1 byte (0-255) numbers)
 - Max: 2^{32} : aprox 4.3×10^9
 - P. ex. 161.116.72.181 (formacio.bq.ub.edu)
 - The 4 levels are hierarchical
- Some addresses are reserved, and some networks are “local”
- (Coming but still not used) IPv6: 128 bits (16 bytes).
Max: 2^{128} (3.4×10^{38})

Name addresses

- IP addresses are not easy. Most hosts have also a “name”:
f. ex. www.ncbi.nlm.nih.edu
- Host names have a structure similar to IP addresses:
 - Top domains (.es, .edu, correspond to full class domains and subnets are indicated by prefixes).
 - ub.edu (161.116.x.x)
 - bq.ub.edu (161.116.72.x)
 - Formacio.bq.ub.edu (161.116.72.181)

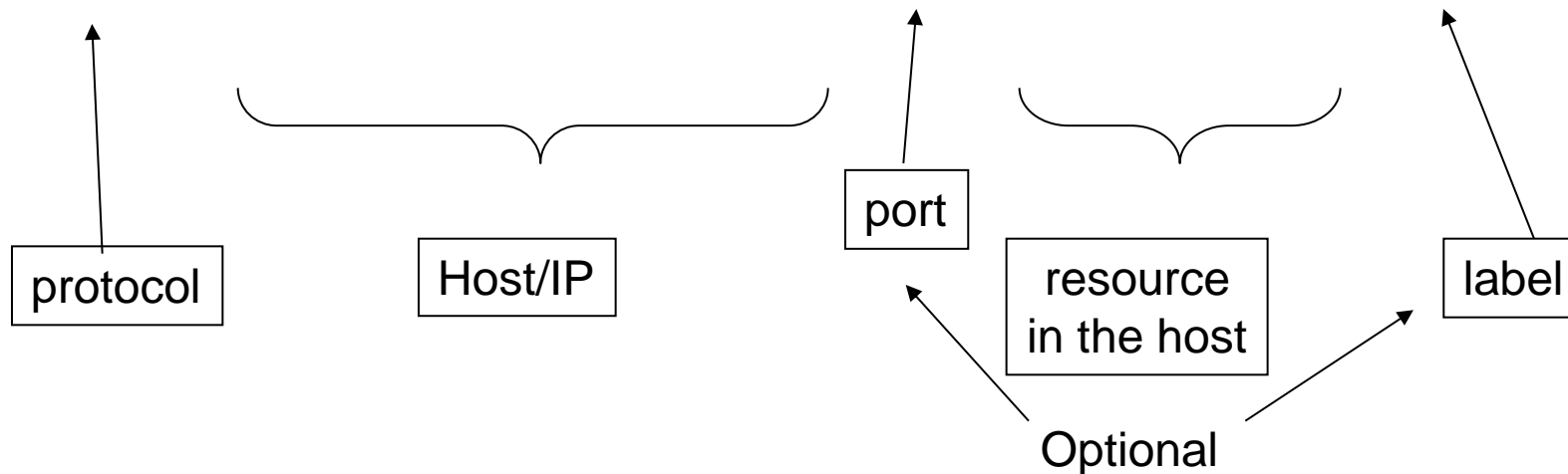
Identifying applications within servers: **Ports**

- Each host has **one (at least) IP address** but has **several ports to identify services within**
- Ports are 2-Byte numbers.
 - 0-1023 **“Well known ports”** (Telnet: 23, FTP: 21, HTTP: 80, HTTPS: 443, ..) (only root)
 - 1024-49151 **“Registered ports”**, usually managed by applications (MySQL: 3306) (only root)
 - 49,152-65,535 **“Dynamic and/or private ports”** freely usable.
- **Communication to ports triggers the specific application** to deal with the data
- However, different ports from the official ones can be used to:
 - Hide applications, Have more than one server in the same IP address, Hide servers in internal networks.

URI/URLs

- Resources must be identified in a way that includes all the necessary details:

`https://formacio.bq.ub.edu:443/index.htm#top`



Missing parts of the URL are added by the client by default!!

Client – server communication (HTTP)

- Most Web Applications use HTTP (hypertext transfer protocol), although sometime FTP, SMTP
- HTTP is a client-server protocol
 - Link between client and server is dynamic
 - Usually **limited to a single transaction**
 - Requests composed by a **query** operation and a variable set of headers (Metadata)
 - Answers: headers + data
- Relevant Operations: GET, POST
 - GET: Simple retrieval, all information included in the URL
 - Simple queries, static information
 - Usable from as hypertext links
 - POST: Upload and retrieval
Query defines the resource, and input data follows
 - PUT: Similar to POST. Used in APIs
- Relevant HTTP headers
 - **Content-type** (POST): input data format
 - **Content-type** (Answer): Data MIME type (text/html, image/jpg, ...)
 - **Location**: Redirects browser
 - **Set-cookie**: Set a “cookie” on users’ software.