

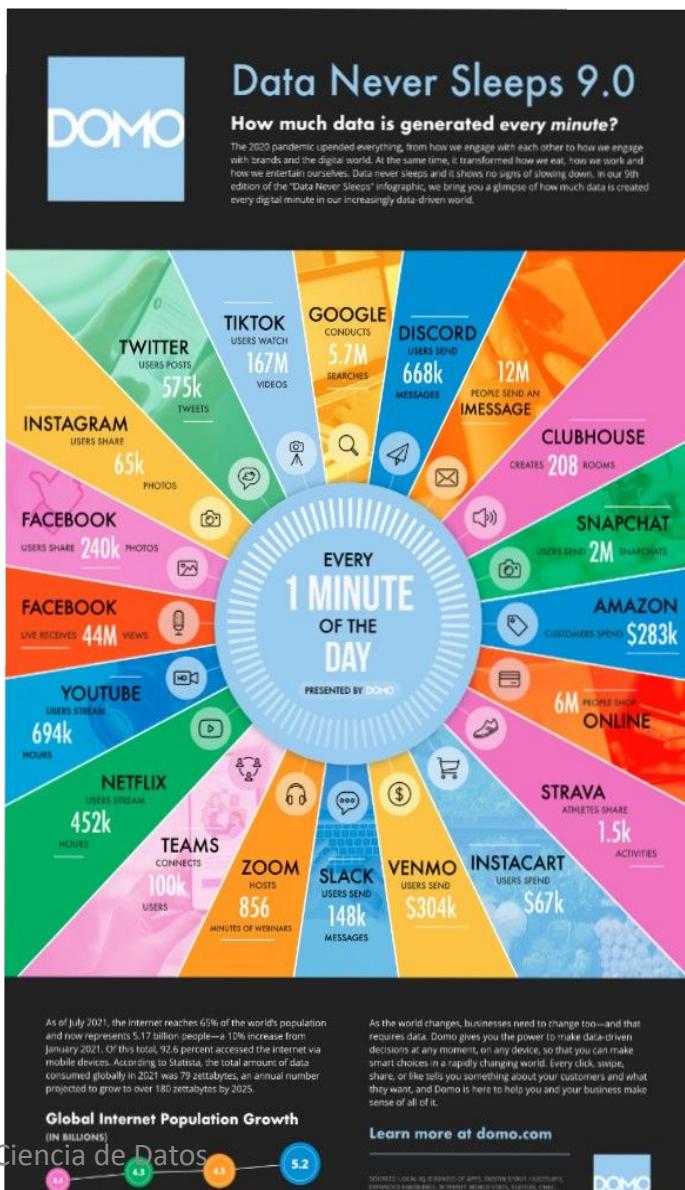
Infraestructuras de computación distribuida

Introducción a los datos masivos y a la Ciencia en abierto
Máster Ciencia de Datos
Aida Palacio Hoz (aidaph@ifca.unican.es)



CSIC
CONSEJO SUPERIOR DE INVESTIGACIONES CIENTÍFICAS



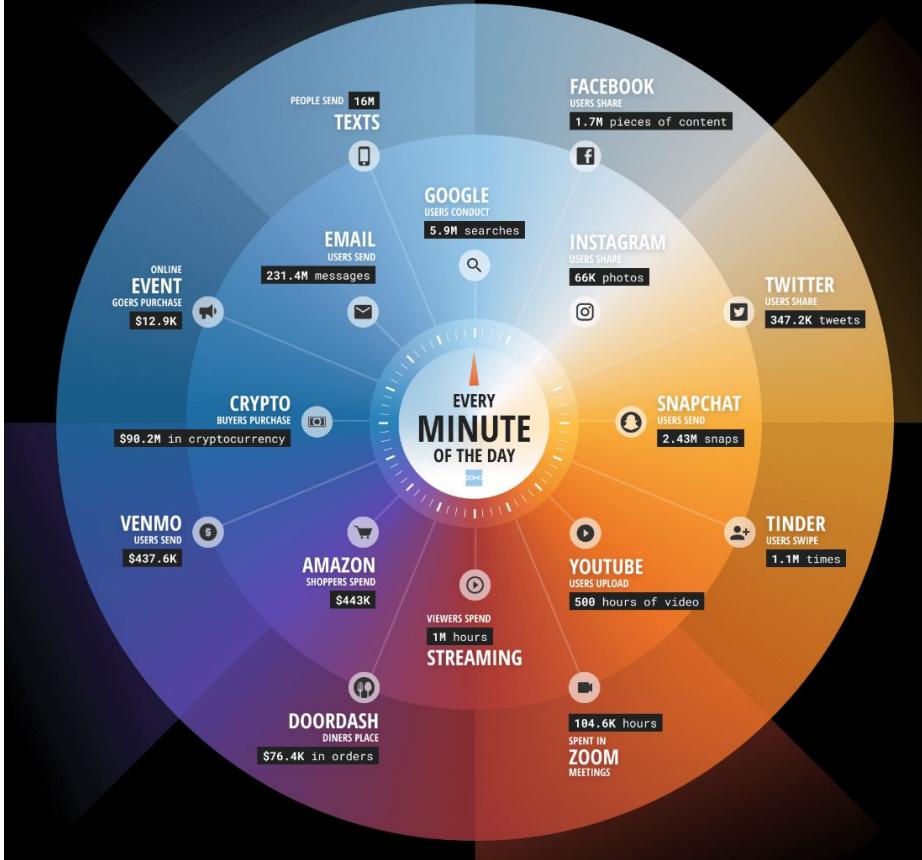




DATA NEVER SLEEPS 10.0

Over the last ten years, digital engagement through social media, streaming content, online purchasing, peer-to-peer payments and other activities has increased hundreds and even thousands of percentage points. While the world has faced a pandemic, economic ups and downs, and global unrest, there has been one constant in society:

our increasing use of new digital tools to support our personal and business needs, from connecting and communicating to conducting transactions and business. In this 10th annual "Data Never Sleeps" infographic, we share a glimpse at just how much data the internet produces each minute from some of this activity, marveling at the volume and variety of information that has been generated.



¿Por qué utilizar computación distribuida en algunos casos de uso?

“The practical difference between obtaining results in hours, rather than weeks or years, is substantial it qualitatively changes the range of studies one can conduct. For example, climate change studies, which simulate thousands of Earth years, are feasible only if the time to simulate a year of climate is a few hours”

President's Information Technology Advisory Committee (PITAC report), 2005

Índice

- Introducción: Modelos de provisión
- Tipos de arquitecturas de computación distribuida
- Infraestructuras de computación: Clúster
 - Ejemplos: Hadoop y Spark
- Tipos de clúster de computación
 - Ejemplo en Supercomputación
 - Ejemplo de ejecución de job en Altamira
 - Grid Computing
 - Cloud Computing
- Refrigeración en un clúster de computación
- BackUp

Introducción

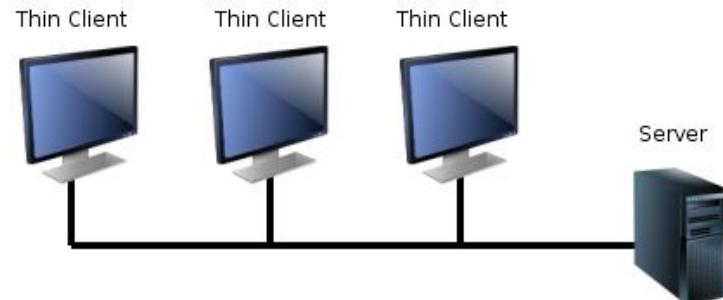
Modelos de provisión

- Computación Centralizada
- Computación Distribuida

Computación Centralizada

La computación centralizada es aquella realizada en una ubicación central, utilizando terminales que se conectan a una computadora central

- Primeros ordenadores muy costosos y ocupaban mucho espacio
- Surgió de la necesidad de que múltiples personas pudieran usar un mismo servidor simultáneamente en un tiempo compartido



Computación Centralizada

- Modelo centralizado: *Mainframe*
 - Desde el comienzo de la “nueva era” computación (1945) hasta 1985
 - Computación y almacenamiento centralizado
 - Acceso mediante terminales de texto o clientes ligeros
 - Seguridad pero muy dependiente de un único dispositivo
 - Poca escalabilidad
 - Muy alto coste



Honeywell Bull DPS 7 en 1990
MasterUniversitario UNED Ciencia de Datos



Cliente ligero



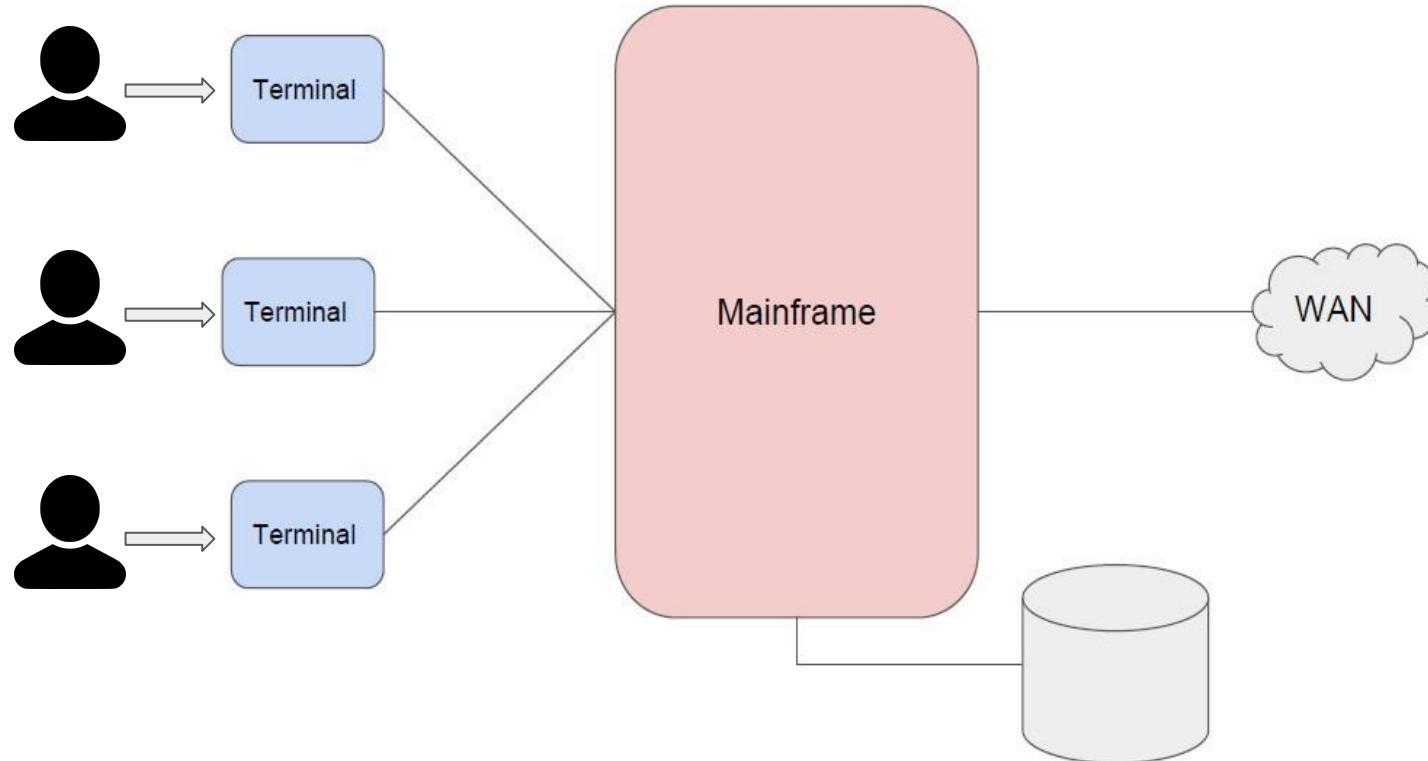
Cliente pesado

Mainframe: Aplicaciones

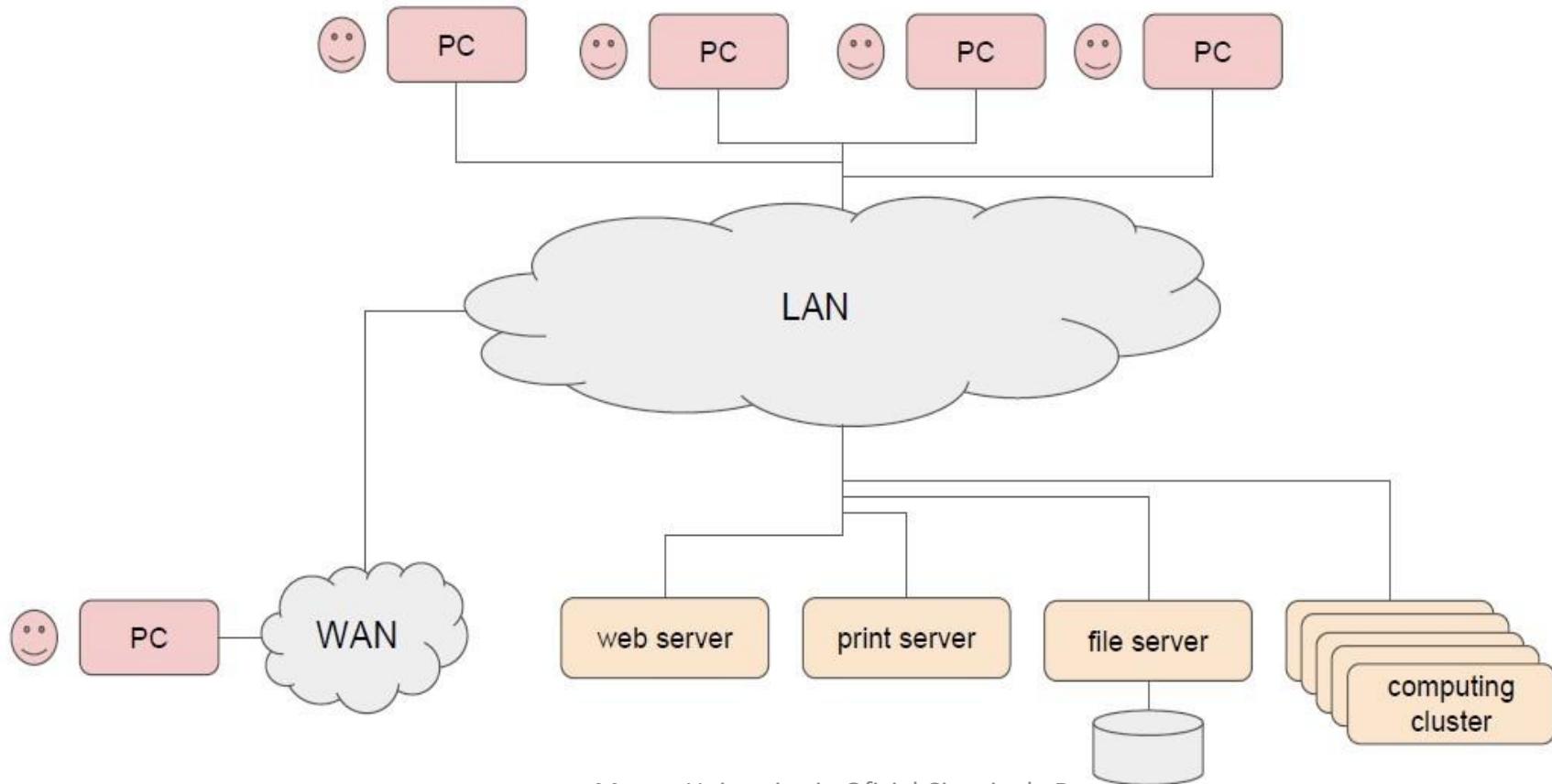
Uso de un ordenador compartido por múltiples personas

- Empresas utilizan clientes ligeros para reducir el coste total de la compra de equipos informáticos
- Actualmente en operaciones críticas
 - Pagos de nóminas
 - Procesamiento de datos masivos:
 - Transacciones de cuentas en los bancos
 - Censos y estadísticas de la industria y el consumidor

Computación Centralizada



Computación Distribuida



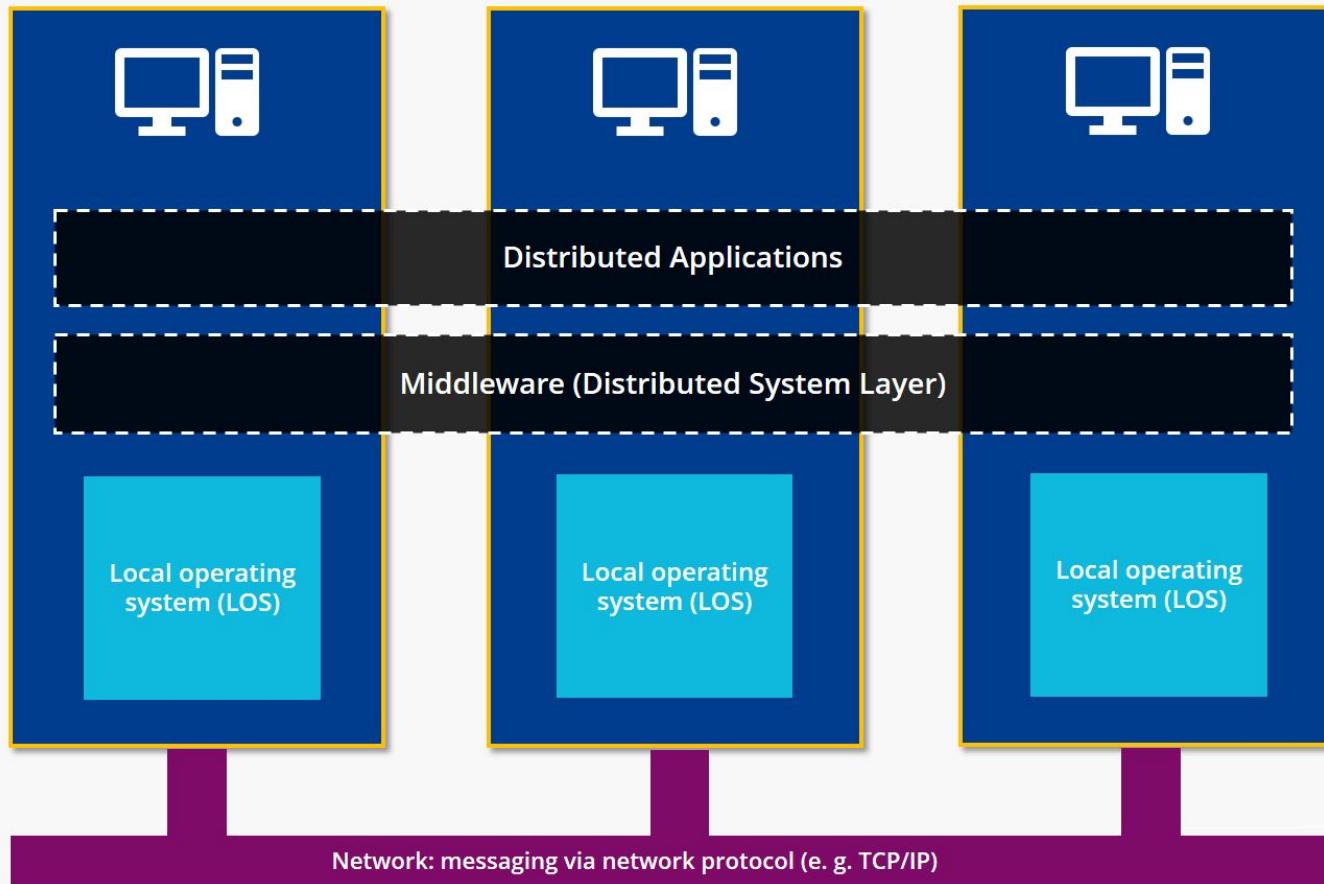
Computación Distribuida

La computación distribuida es un modelo computacional diseñado para resolver problemas utilizando un gran número de ordenadores organizados en una red de telecomunicaciones

En computación distribuida, cada problema se divide en múltiples tareas, que se resuelven en uno o más ordenadores: **Sistema distribuido**

- Múltiples CPUs ubicadas en distinto servidor trabajando conjuntamente
- Colección de computadores independientes que se comportan como un sistema coherente, comunicándose a través de una red -> Intercambio de mensajes
- El **objetivo** de un sistema de computación distribuida es calcular y generar un resultado de una función matemática tras recibir unos datos de entrada

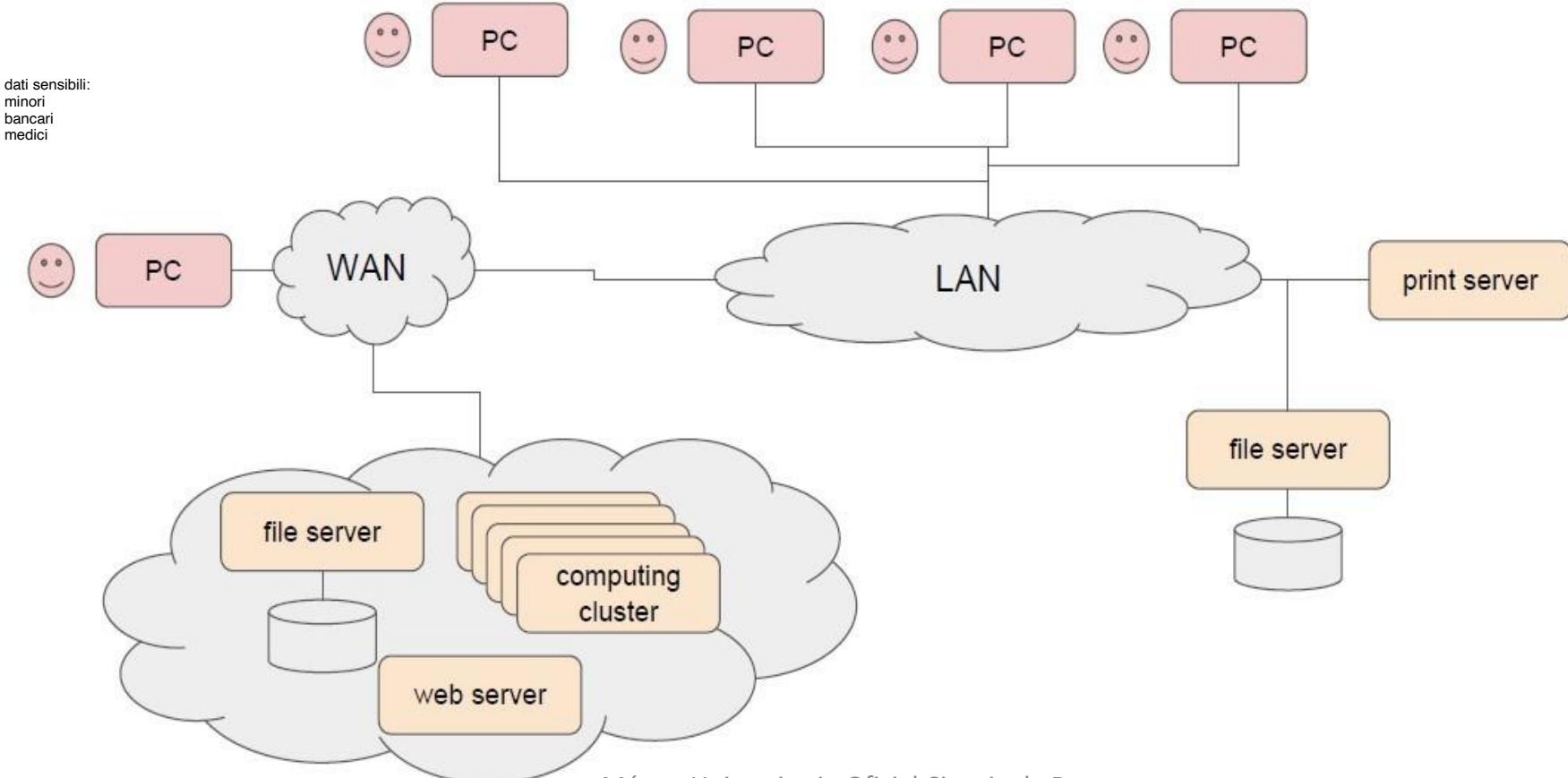
Distributed Computing



Computación Distribuida

- Características:
 - Alto rendimiento, alta disponibilidad, balanceo de carga, escalabilidad
 - Su instalación hardware es más fácil y económica gracias a su flexibilidad
 - Interacción de componentes para conseguir un objetivo común
- Ventajas:
 - Tolerancia a fallos, fiabilidad, escalabilidad, disponibilidad, menor coste
- Desventajas:
 - Seguridad, privacidad, seguridad de la red
 - Complejidad a la hora de configurarlos, mantenerlos y trabajar sobre ellos

Computación distribuida: Sistema Cloud

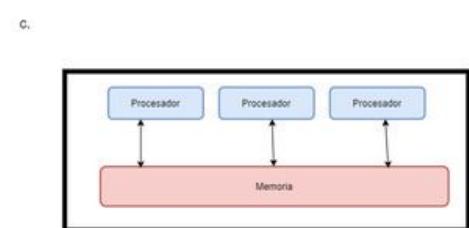
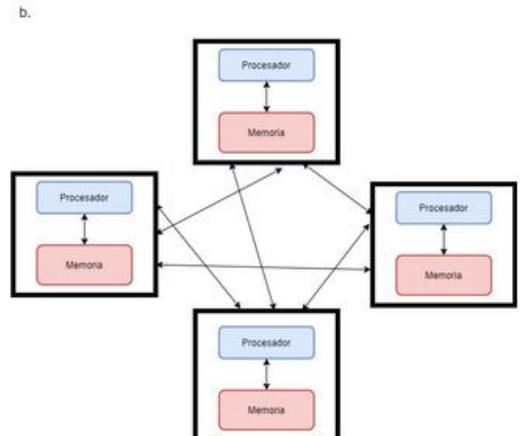


Computación Distribuida: Aplicaciones

- Cliente/Servidor
 - Servidores commodity y PCs convencionales
 - Coste incremental
- Sistemas Cloud
 - Computación centralizada en computadores commodity
 - Acceso a través de la red
 - Alta capacidad para escalar
 - Modelos de negocio: pago por uso
- Red corporativa (servidores web, impresoras, servidor correo electrónico)
- Clúster de computación

Computación Distribuida vs Paralelismo

- Los sistemas distribuidos son una subclase de los sistemas paralelos
- Los sistemas paralelos se centran en el aumento del rendimiento, mientras que los sistemas distribuidos se centran en la tolerancia de fallos parciales
- En la computación paralela, todos los procesadores pueden tener acceso a una memoria compartida para intercambiar información entre ellos y en la computación distribuida, cada procesador tiene su propia memoria privada, donde la información se intercambia pasando mensajes entre los procesadores

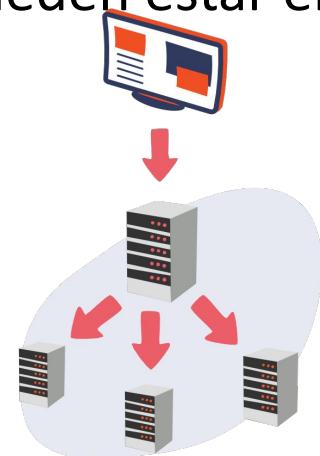


Tipos de Arquitecturas de computación distribuida

- Cliente-Servidor (Master - Slave)
- Peer-to-peer (P2P)

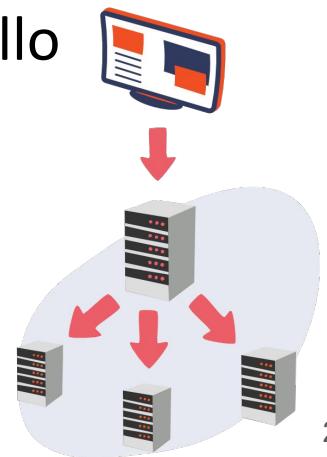
Arquitecturas: Cliente-Servidor

- Sistema **centralizado de organización jerárquica**: La carga de trabajo se distribuye entre un **servidor** (software proveedor del servicio) y un **cliente** (software demandante)
 - Servidor (master): Realizar una carga a petición de un cliente
 - Cliente (slave): Demandar un servicio de un servidor
- Normalmente se comunican a través de la red, pero pueden estar en la misma máquina
- Ejemplos:
 - Servidor web y navegador
 - Webmail y servidor de correo
 - Plataforma de streaming y navegador



Arquitecturas: Cliente-Servidor

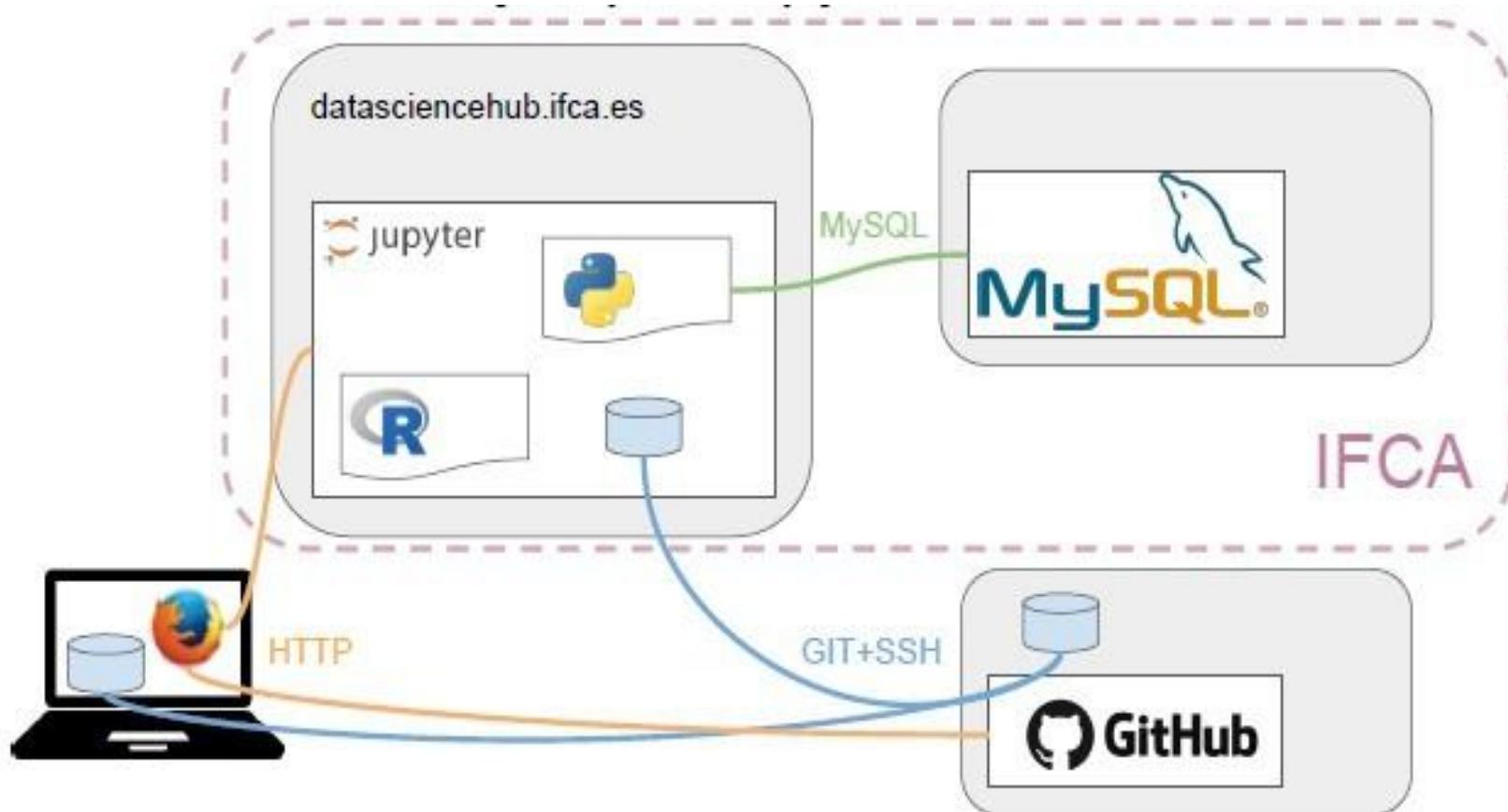
- Ventajas:
 - escalabilidad: añadir más nodos “slave” a la red
 - tolerancia a fallos: en caso de que un “slave” falle, se reparte el trabajo entre el resto de clientes
- Inconvenientes:
 - Si solo hay un máster -> único punto de contacto/ fallo
 - Si el máster falla, todo el sistema falla
- Práctica común: múltiples masters (HA)



Cliente-servidor: Ejemplos

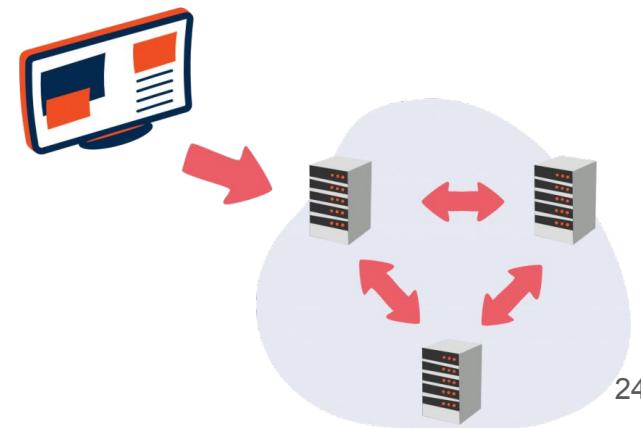
- Ejemplos de servidor
 - SQL: MySQL, PostgreSQL, MariaDB
 - HTTP (servidor web): Apache, Nginx, Microsoft IIS, LightHTTPD
 - DNS: Bind
- Ejemplos de clientes:
 - SQL
 - Clientes: MySQL, PostgreSQL, MariaDB, phpmyadmin
 - Librerías: Python-sqlalchemy
 - HTTP:
 - Navegadores web: Firefox, Chrome
 - Librerías: Python-requests, cURL

Cliente-servidor: Ejemplo Jupyterhub



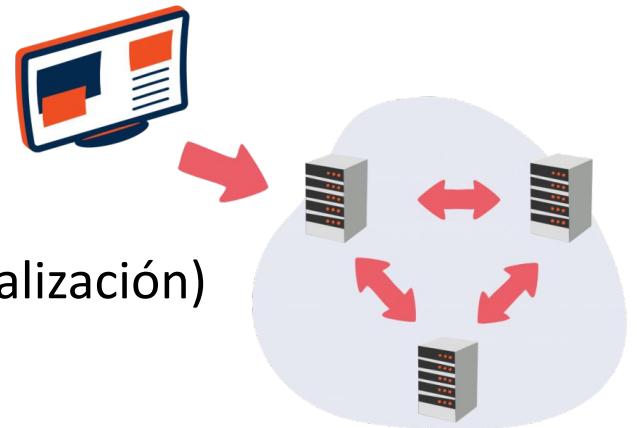
Arquitecturas: Peer-to-Peer (P2P)

- Sistema **descentralizado ya que no existen diferencias entre los nodos de la red**: La carga de trabajo se distribuye entre los diferentes componentes de la red (peers)
 - Arquitectura descentralizada: No tiene clientes ni servidores fijos
 - Todos los nodos tienen el mismo rol en la red
 - Evitar que se produzca un cuello de botella en el nodo central
- Ejemplos:
 - Redes multimedia
 - BitTorrent/qBittorrent
 - Bitcoin

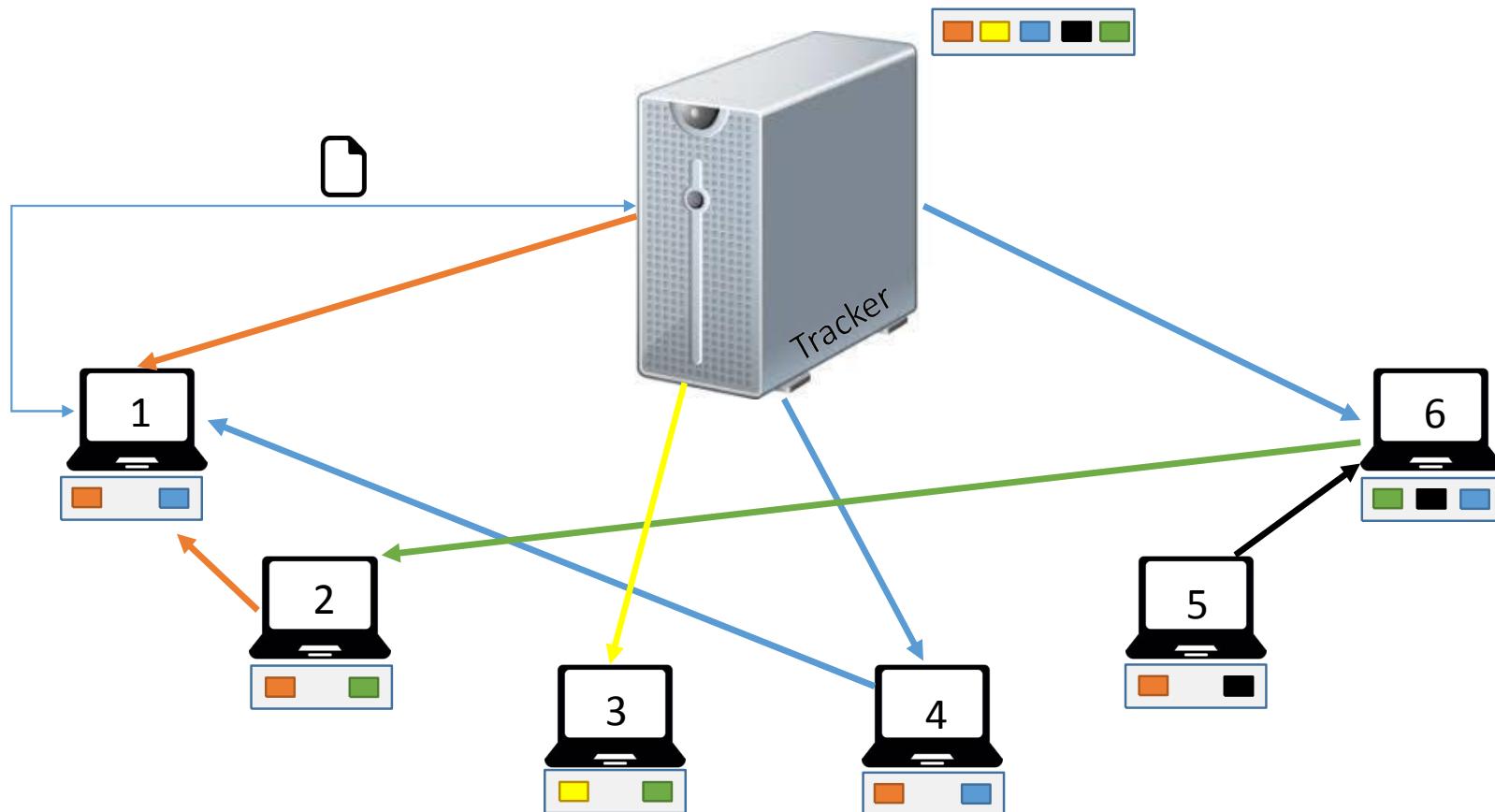


Peer-to-peer (P2P)

- Características:
 - No existen nodos con funciones especiales: ningún nodo es imprescindible para el funcionamiento de la red.
 - Todos los nodos se tratan como iguales
- Ventajas:
 - Tolerancia a fallos, escalabilidad, disponibilidad
 - No hay un punto de contacto de fallo (descentralización)
- Desventajas:
 - Seguridad, privacidad, seguridad de la red
 - Sistema de toma de decisiones más complejo al no haber jerarquía
 - P2P centralizado: Punto de fallo central



Arquitectura P2P: Ejemplo BitTorrent



Arquitectura P2P: Ejemplo BitTorrent

- Ventajas:
 - No hay saturación de un servidor central
 - Redundancia: Siempre que un peer haya recibido el archivo completo
 - Alta disponibilidad de descarga: Muchos usuarios conectados pudiendo descargar fragmentos de varios de ellos a la vez
- Desventajas:
 - Errores en los ficheros a la hora de componer el fichero completo
 - Alta dependencia de los peers: Si los usuarios que reciben el archivo completo se desconectan nada más descargar el fichero completo
 - Dependencia del tracker

Infraestructuras de computación

Clúster de computación

Un **clúster** es un tipo de **sistema distribuido** que consiste en una colección de servidores **interconectados entre sí (habitualmente por una red de alta velocidad)** para trabajar como un único servidor como recurso de computación integrado

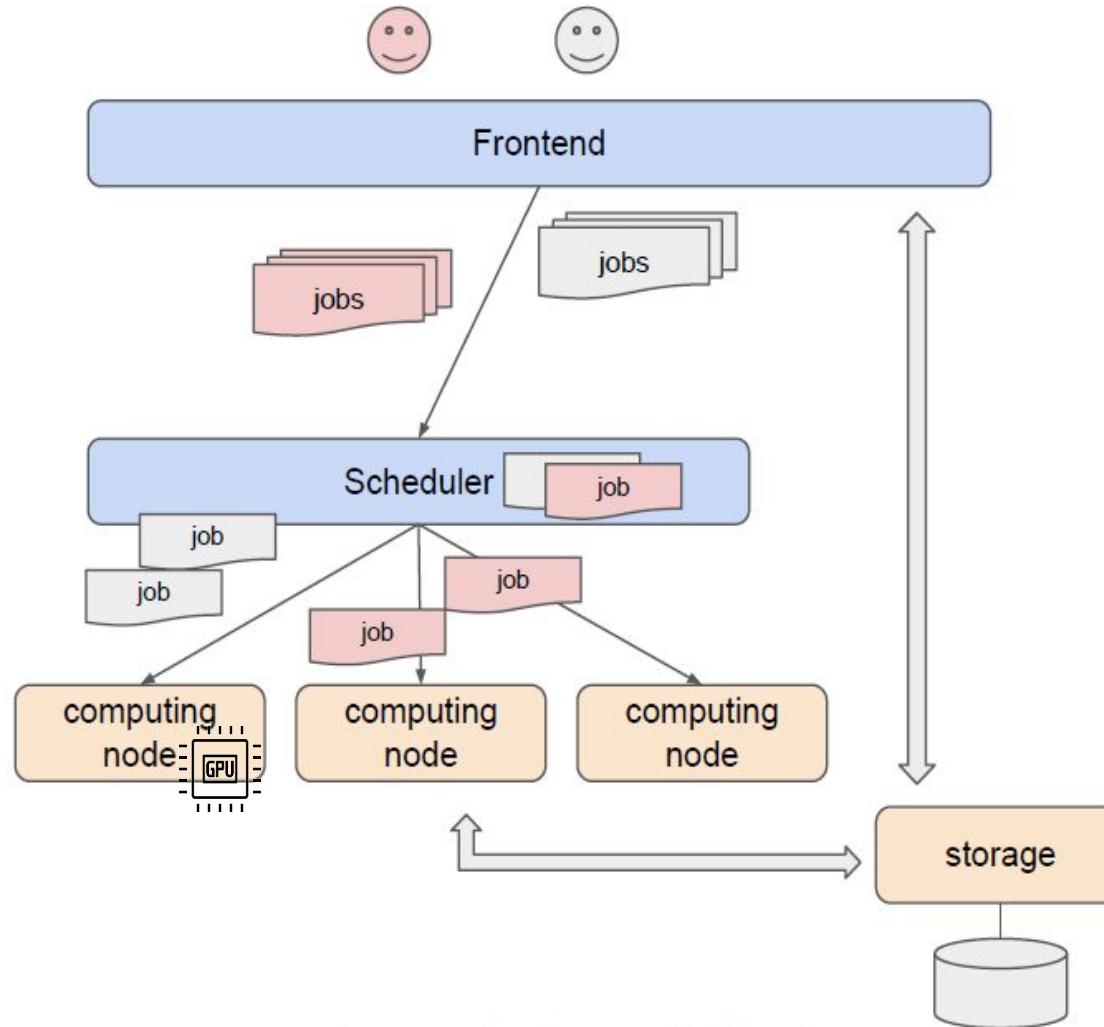
- ¿Para qué sirven?
 - Obtener mayor capacidad de cómputo
 - Menor precio que un teórico único sistema equivalente
- Constan de una capa de gestión que mantiene los diferentes nodos y distribuye la carga entre ellos

Clúster de computación

- Tipos:
 - single image, cluster computación (HPC, HTC), load-balancing, high-availability (HA), etc
- Características:
 - Alto rendimiento, alta disponibilidad, balanceo de carga, escalabilidad
 - Flexibilidad: clúster homogéneo, semi homogéneo, heterogéneo
 - Sistema de gestión del clúster: interacción con el usuario y procesos que corren en él para optimizar el funcionamiento

Componentes clúster de computación

- Frontend (nodos de acceso)
- Nodos de cálculo
- Interconexión de red (internet + datos)
- Almacenamiento distribuido
- Protocolos de comunicación y servicios
- Aplicaciones
- Sistema gestor de colas
 - Prioriza peticiones de usuarios
 - Gestiona los recursos



Clúster de computación: Aplicaciones

- Los clústeres pueden también clasificar en:
 - clusters comerciales (HA o HTC)
 - clústeres científicos (HPC o HTC)
- Realizar tareas de cálculo inabarcables con un único computador
- Investigación:
 - Simulación de un problema
 - Análisis de datos masivos
- Industria
 - Diseño de nuevos productos
 - Defensa, aplicaciones militares
 - Finanzas

Problemas de la tecnología tradicional

1. Almacenamiento masivo de datos: No es posible con un sistema tradicional de un único nodo
2. Almacenamientos de datos heterogéneos: Big Data incluye datos sin formato
 - a. Datos estructurados
 - b. No estructurados
 - c. Semiestructurados
3. Acceso y velocidad de procesamiento: Aumenta la capacidad del disco duro pero no la velocidad de transferencia o acceso
 - Single-node: 1TB de datos y red de 100Mbps -> 2.91 horas
 - 4-nodes: 43 min.



Ejemplo clúster: Hadoop

Hadoop es una plataforma *open-source* que permite almacenar y procesar grandes cantidades de datos (Big Data) estructurados, semiestructurados y no estructurados en un entorno distribuido

- Nace como iniciativa de Apache para dar soporte al paradigma de programación MapReduce inicialmente publicado por Google
- Gestionado y mantenido por Apache Software Foundation (ASF)
- Gran escalabilidad: despliegue en single-mode y en miles de máquinas
- Alta fiabilidad
- Cada nodo ofrece capacidad de cómputo y almacenamiento local
- Aplicaciones:
 - Marketing digital: promociones o campañas mucho más directas y menos costosas
 - Finanzas: Mejorar la eficiencia operativa y gestionar el riesgo en la toma de decisiones
 - Data Science: Inteligencia Artificial, Internet of Things (IoT)



Hadoop Distributed Filesystem (HDFS)

- HDFS es el componente principal de Hadoop
- Se trata del sistema de ficheros por excelencia en Big Data para almacenar datasets con tipos de datos estructurados, semiestructurados y sin formato
- Basado en Java
- Está optimizado para almacenar grandes cantidades de datos y mantener varias copias en el clúster para garantizar una alta disponibilidad y tolerancia a fallos.
- Proporciona una capa de abstracción entre todos los nodos disponibles
 - FS Global
 - Algoritmo de posicionamiento y replicado de los datos
 - Modificar el procesamiento es más óptimo que mover los datos
 - Incrementar la capacidad de hadoop es trivial
 - Incrementando el número de nodos
 - Incrementando el número de discos
- Los ficheros almacenados están divididos en bloques de un mismo tamaño (128MB configurable) y se distribuyen en los nodos que forman el clúster



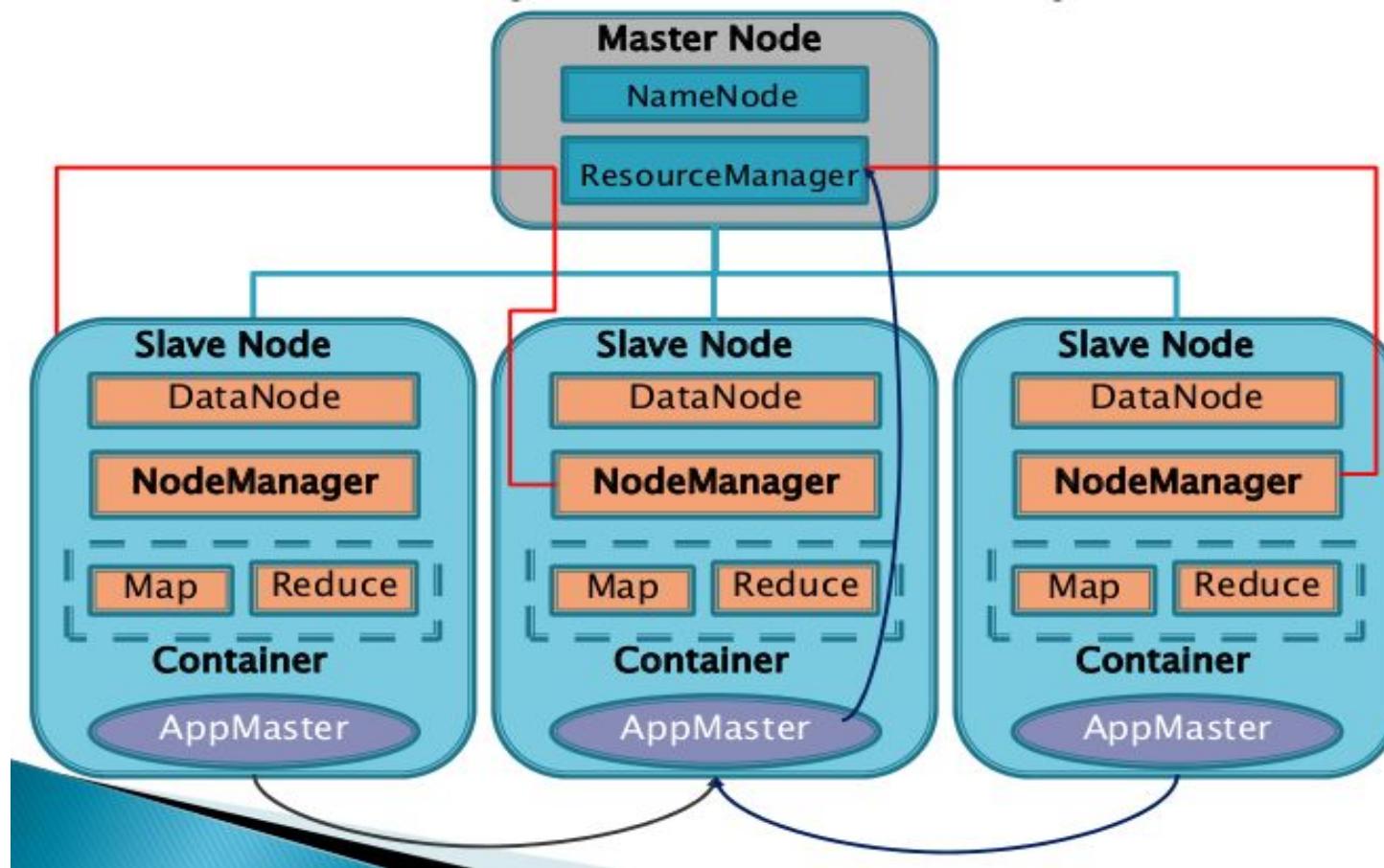
MapReduce

- MapReduce es un paradigma de procesamiento de datos en dos pasos que se ejecutan de manera distribuida en diferentes nodos de procesamiento (DataNodes)
 - Map: Se generan los pares clave/valor, filtrando, agrupando y/o transformando los datos
 - Reduce: genera el fichero de salida gestionando la agregación de los valores obtenidos en la primer fase Map
- Para poder empezar la fase de reduce, la fase map debe haber finalizado
- El ejemplo más común de MapReduce es el Word Count: Consiste en contar las veces que una palabra aparece en un conjunto de documentos.

Yet Another Resource Negotiator (YARN)

- Framework que permite a Hadoop soportar diversos motores de ejecución como MapReduce
- Proporciona un *scheduler* de los trabajos que envían los usuarios a ejecutar al clúster de Hadoop
 - Gestión de recursos
 - Planificación
 - Monitorización
- Componentes:
 - **Resource Manager:** Encargado de gestionar la distribución de los recursos del cùster de Yarn
 - **NodeManager:** Gestiona los trabajos enviados por el RM y proporciona los recursos de cómputo necesarios para ejecutar las aplicaciones encapsulándolo en contenedores
 - **Application Manager:** Responsable de negociar los recursos apropiados con el RM y monitorizar su estado y progreso

Arquitectura de Hadoop



- **Spark:** Paradigma de procesamiento distribuido de datos

- Mejora el paradigma MapReduce aumentando la velocidad en 100 veces
- Trabaja con memoria en vez de disco
- Soporta aplicaciones ETL (extract, Transform, Load), Machine Learning y Streaming de datos



- **Zookeeper:** Servicio para la coordinación de procesos distribuidos

- **Tez:**

- Framework de programación de flujo de datos
- Evolución de MapReduce que ejecuta sobre Yarn optimizando el código para mejorar el rendimiento en 10 veces



- **Oozie:** Permite ejecutar y planificar en el tiempo trabajos y tareas en Hadoop mediante configuraciones XML

- Soporta varios tipos de Jobs: MapReduce, Pig, Hive, scripts de shell o programas Java

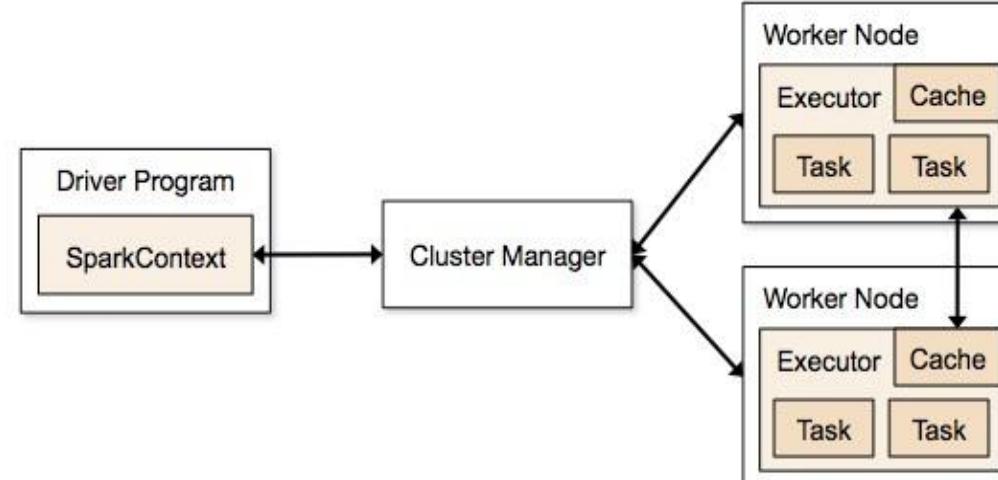
- **Pig:** Plataforma de alto nivel para crear programas MapReduce

- Utiliza el lenguaje llamado Pig Latin



Ejemplo de clúster de computación 2: Apache Spark

- **Framework de computación distribuida** muy popular en Big Data
- Aplicaciones en Medicina, Finanzas, multimedia, ciberseguridad, etc
- Spark suele utilizar Apache Hadoop como almacenamiento de datos
- **Arquitectura master-slave**
- Provee una sencilla serie de APIs para procesar paralelamente grandes volúmenes de datos mientras el usuario no necesita tener gran conocimiento sobre cómo se gestionan los recursos

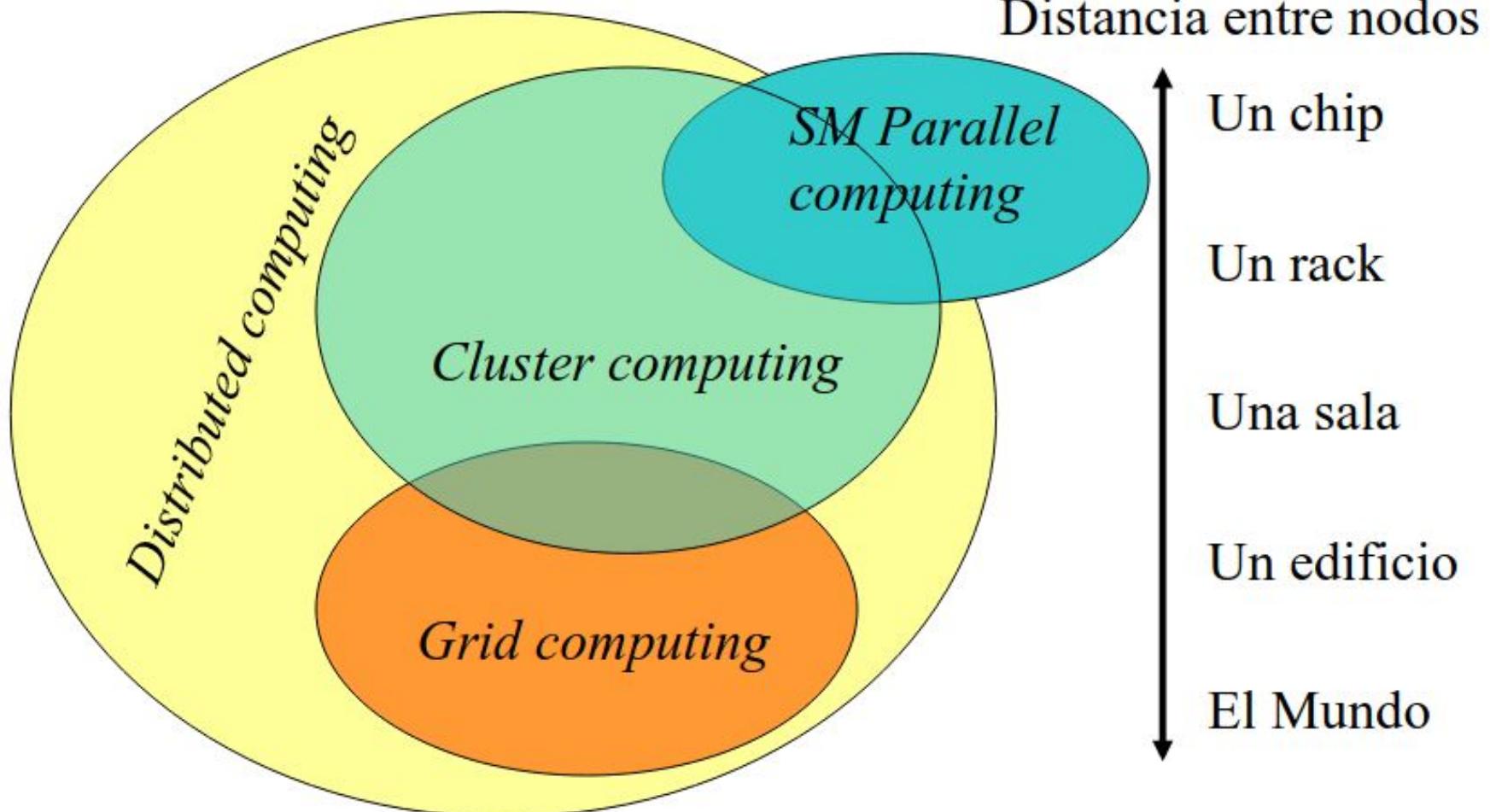


Tipos de clúster de computación

- High Performance Computing (HPC)
- Grid Computing
- Cloud Computing

Tipos clúster de computación

- **HPC: *High Performance Computing***
 - Ejecución de tareas *tightly coupled* (es decir, con alto grado de paralelismo)
 - Ejecución de tareas con gran comunicación de datos, gran capacidad computacional, grandes cantidades de memoria
 - Comprometen los recursos del clúster grandes períodos de tiempo
- **HAC: *High Availability Computing***
 - Alta disponibilidad de los servicios que ofrecen
 - Alta confiabilidad: gran tolerancia a fallos
- **HTC: *High Throughput Computing (Alta eficiencia)***
 - Enfocados a ejecutar tareas con pocas dependencias (*loosely coupled*)
 - Ejecución de muchas tareas que no requieren gran capacidad de cómputo



Tipos de clúster de computación

- **High Performance Computing (HPC)**
- Grid Computing
- Cloud Computing

High Performance Computing (HPC)

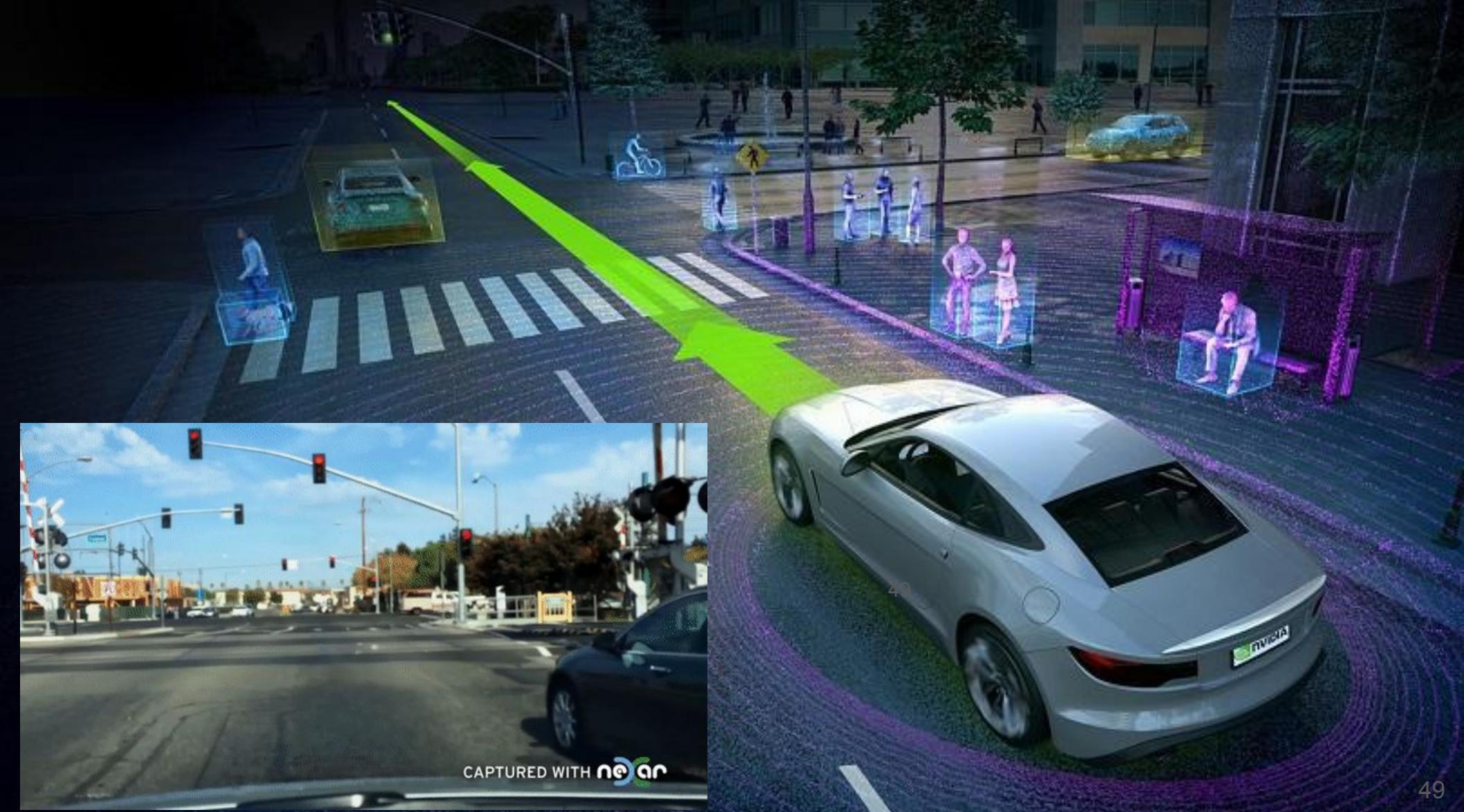
= Supercomputación

- **Alto rendimiento:** consiste en ejecutar tareas que requieren grandes capacidades de cómputo: cpu y memoria.
- Estas tareas pueden ejecutarse durante grandes periodos de tiempo
- Sistemas donde la cpu está optimizada para realizar paralelismo

¿Qué es un supercomputador?



Aplicaciones en Supercomputación

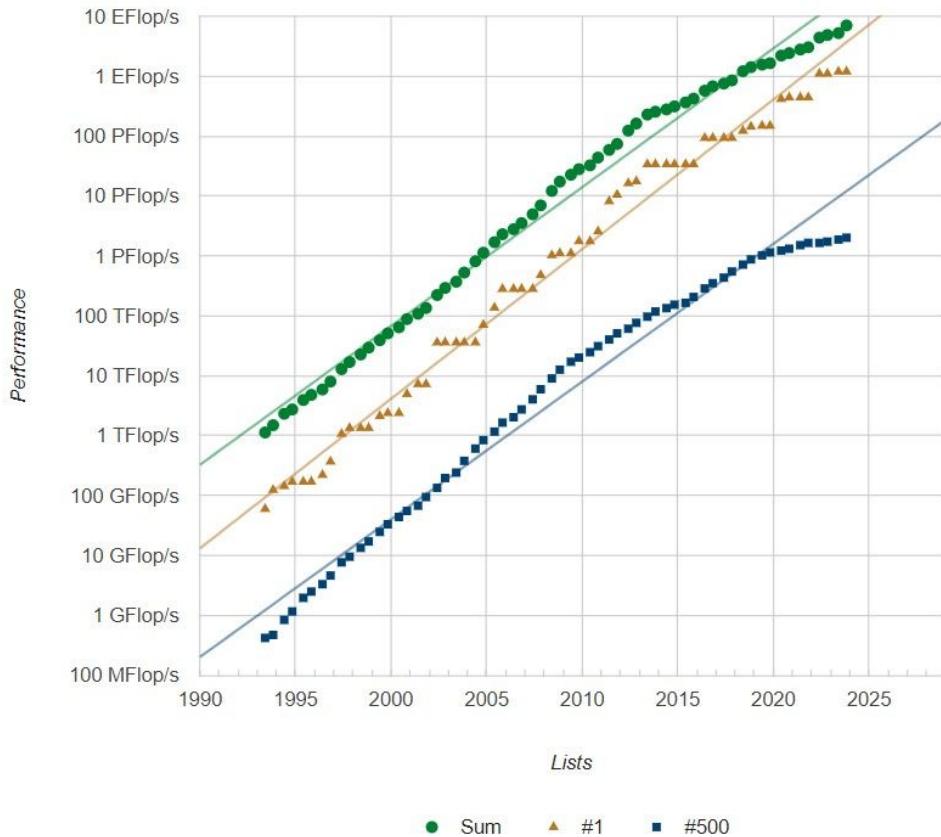


SpiNNaker



Lista TOP 500

- Los 500 supercomputadores más potentes del mundo
- Esta lista comenzó en 1992 y publica 2 veces al año
- **Objetivo:** facilitar colaboraciones entre instituciones, intercambio de datos, software y aplicaciones usadas por un ordenador
- Ordenados por el rendimiento ejecutando el benchmark LINPACK



Lista TOP 500: Evolución



June 1993

CM-5 (59,7 GigaFlops)



TODAY



Frontier (1.206 PetaFlops)

Lista TOP 500: Junio 2023



Rank	System	Cores	Rmax (PFlop/s)	Rpeak (PFlop/s)	Power (kW)
1	Frontier - HPE Cray EX235a, AMD Optimized 3rd Generation EPYC 64C 2GHz, AMD Instinct MI250X, Slingshot-11, HPE DOE/SC/Oak Ridge National Laboratory United States	8,730,112	1,102.00	1,685.65	21,100
2	Supercomputer Fugaku - Supercomputer Fugaku, A64FX 48C 2.2GHz, Tofu interconnect D, Fujitsu RIKEN Center for Computational Science Japan	7,630,848	442.01	537.21	29,899
3	LUMI - HPE Cray EX235a, AMD Optimized 3rd Generation EPYC 64C 2GHz, AMD Instinct MI250X, Slingshot-11, HPE EuroHPC/CSC Finland	1,110,144	151.90	214.35	2,942
98	MareNostrum - Lenovo SD530, Xeon Platinum 8160 24C 2.1GHz, Intel Omni-Path, Lenovo Barcelona Supercomputing Center Spain	153,216	6.47	10.30	1,632
42 en 2020					

Lista TOP 500: Junio 2024

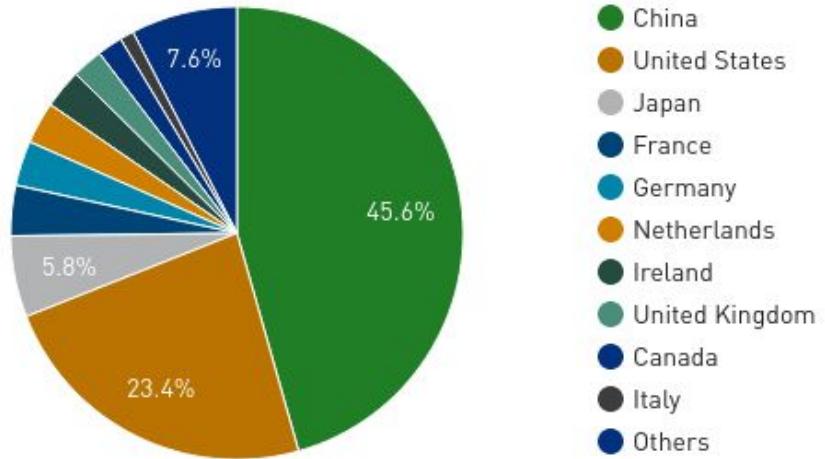
Rank	System	Cores	Rmax (PFlop/s)	Rpeak (PFlop/s)	Power (kW)
1	Frontier - HPE Cray EX235a, AMD Optimized 3rd Generation EPYC 64C 2GHz, AMD Instinct MI250X, Slingshot-11, HPE DOE/SC/Oak Ridge National Laboratory United States	8,699,904	1,206.00	1,714.81	22,786
2	Aurora - HPE Cray EX - Intel Exascale Compute Blade, Xeon CPU Max 9470 52C 2.4GHz, Intel Data Center GPU Max, Slingshot-11, Intel DOE/SC/Argonne National Laboratory United States	9,264,128	1,012.00	1,980.01	38,698
3	Eagle - Microsoft NDv5, Xeon Platinum 8480C 48C 2GHz, NVIDIA H100, NVIDIA Infiniband NDR, Microsoft Azure Microsoft Azure United States	2,073,600	561.20	846.84	
4	Supercomputer Fugaku - Supercomputer Fugaku, A64FX 48C 2.2GHz, Tofu interconnect D, Fujitsu RIKEN Center for Computational Science Japan	7,630,848	442.01	537.21	29,899
5	LUMI - HPE Cray EX235a, AMD Optimized 3rd Generation EPYC 64C 2GHz, AMD Instinct MI250X, Slingshot-11, HPE EuroHPC/CSC Finland	2,752,704	379.70	531.51	7,107

Lista TOP 500: Junio 2024 (Spain)



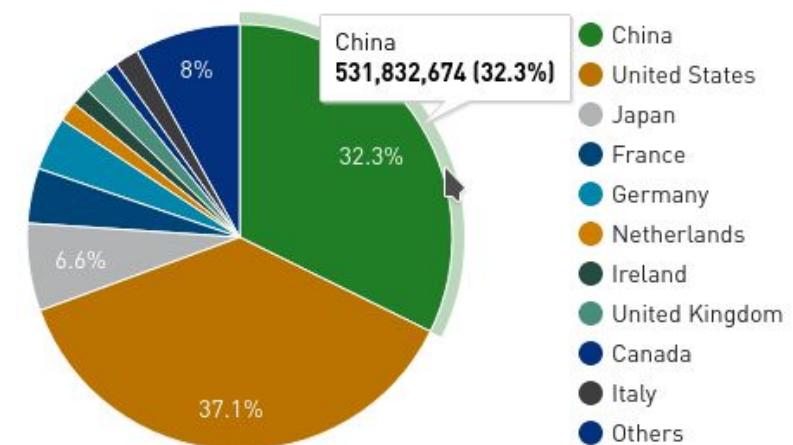
Rank	System	Cores	Rmax (PFlop/s)	Rpeak (PFlop/s)	Power (kW)
8	MareNostrum 5 ACC - BullSequana XH3000, Xeon Platinum 8460Y+ 32C 2.3GHz, NVIDIA H100 64GB, Infiniband NDR, EVIDEN EuroHPC/BSC Spain	663,040	175.30	249.44	4,159
<hr/>					
22	MareNostrum 5 GPP - ThinkSystem SD650 v3, Xeon Platinum 03H-LC 56C 1.7GHz, Infiniband NDR200, Lenovo EuroHPC/BSC Spain	725,760	40.10	46.37	5,753
<hr/>					
144	MareNostrum - Lenovo SD530, Xeon Platinum 8160 24C 2.1GHz, Intel Omni-Path, Lenovo Barcelona Supercomputing Center Spain	153,216	6.47	10.30	1,632
<hr/>					

Countries System Share



- China
- United States
- Japan
- France
- Germany
- Netherlands
- Ireland
- United Kingdom
- Canada
- Italy
- Others

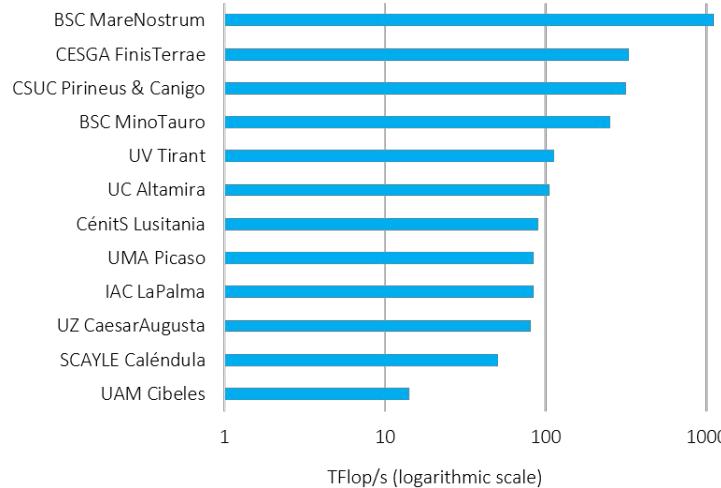
Countries Performance Share



- China
- United States
- Japan
- France
- Germany
- Netherlands
- Ireland
- United Kingdom
- Canada
- Italy
- Others

Red Española de Supercomputación (RES)

Red Española de Supercomputación (2007): compuesta de 14 nodos para ofrecer los recursos necesarios para el desarrollo de proyectos científicos y tecnológicos innovadores y de alta calidad



GREEN500



- Los 500 ordenadores más eficientes energéticamente
- Primera publicación en 2007
- Supercomputadores medidos por rendimiento-por-vatio (FLOPS-per-watt)

Rank	TOP500 Rank	System	Cores	Rmax (PFlop/s)	Power (kW)	Energy Efficiency (GFlops/watts)
1	189	JEDI - BullSequana XH3000, Grace Hopper Superchip 72C 3GHz, NVIDIA GH200 Superchip, Quad-Rail NVIDIA InfiniBand NDR200, ParTec/EVIDEN EuroHPC/FZJ Germany	19,584	4.50	67	72.733
2	128	Isambard-AI phase 1 - HPE Cray EX254n, NVIDIA Grace 72C 3.1GHz, NVIDIA GH200 Superchip, Slingshot-11, HPE University of Bristol United Kingdom	34,272	15	8	MareNostrum 5 ACC - BullSequana XH3000, Xeon Platinum 8460Y+ 32C 2.3GHz, NVIDIA H100 64GB, Infiniband NDR, EVIDEN EuroHPC/BSC Spain
13	1	Frontier - HPE Cray EX235a, AMD Optimized 3rd Generation EPYC 64C 2GHz, AMD Instinct MI250X, Slingshot-11, HPE DOE/SC/Oak Ridge National Laboratory United States	8,699,904	1,206.00	22,786	52.927

Fugaku

Nº1 Top 500 Desde junio 2020 hasta junio 2022

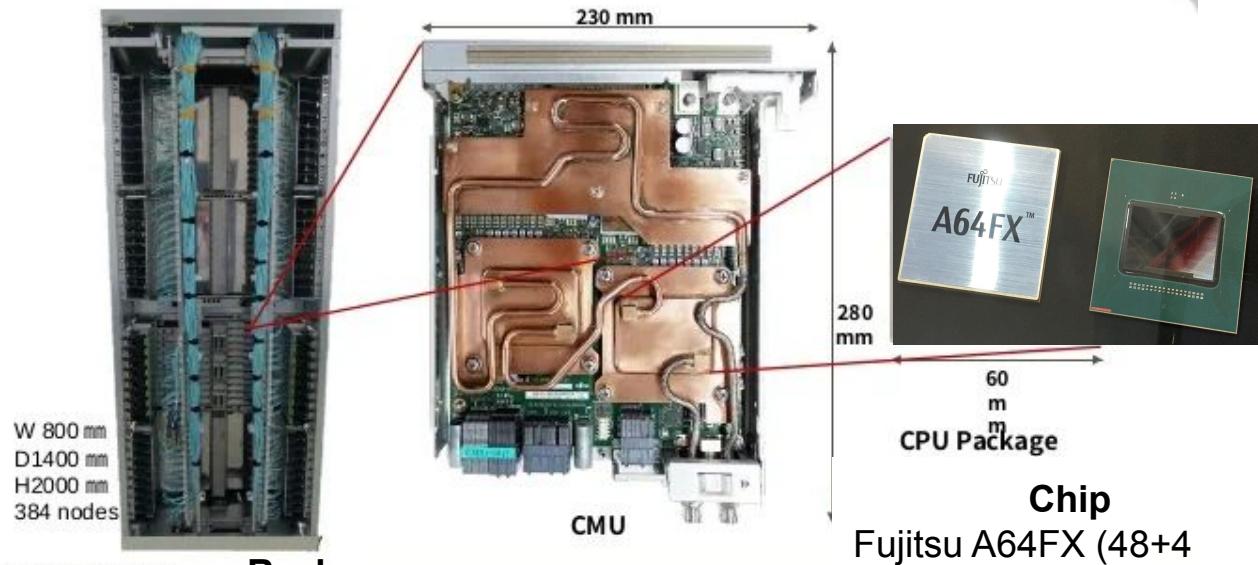


- System**
- 416 PetaFLOPS
 - Total 158,976 nodos
 - > 400 racks
 - 150 PB almacenamiento
 - SO Linux
 - 1000mill. \$ EEUU



Fugaku Chassis, PCB (w/DLC), and CPU Package

FUJITSU



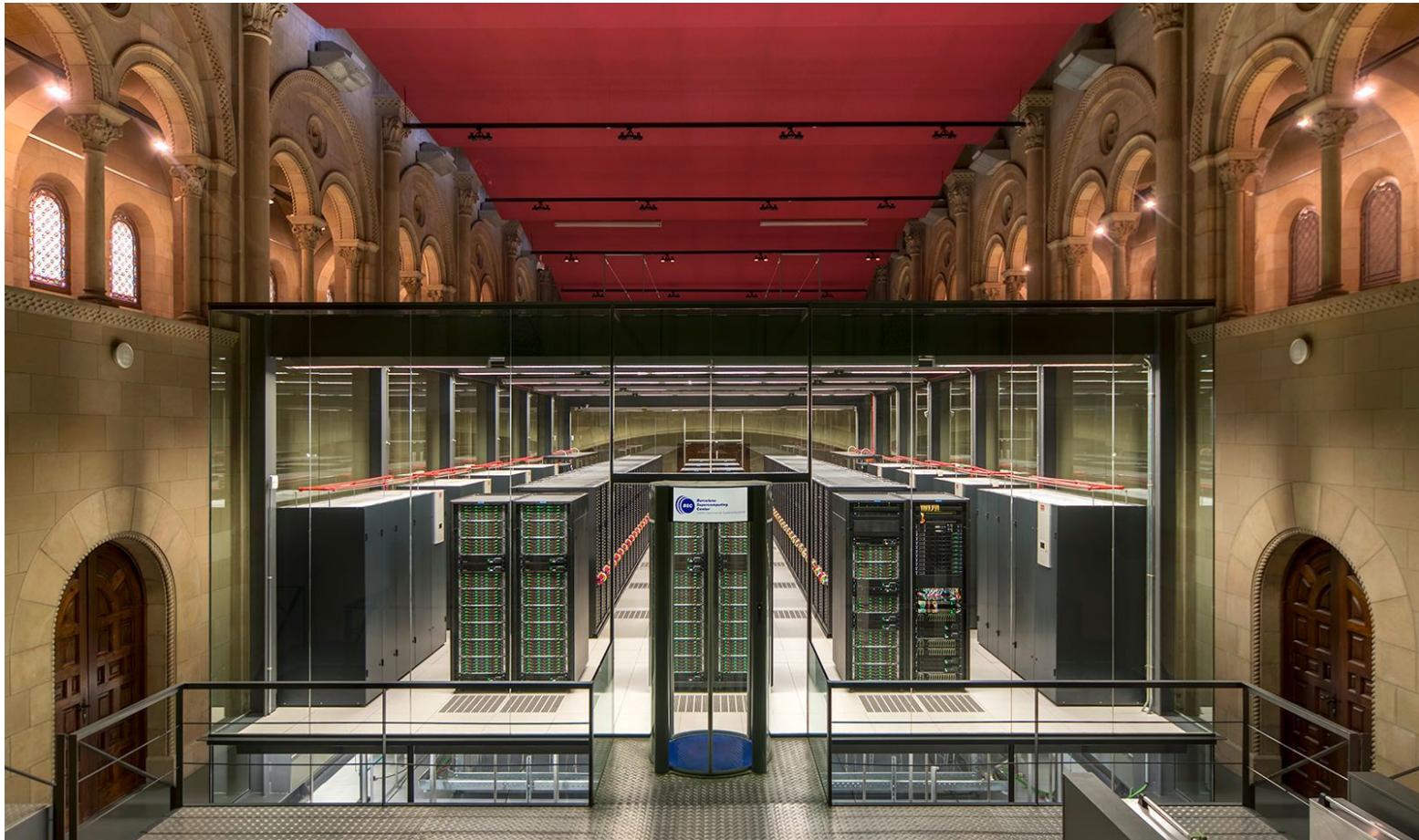
Chip
Fujitsu A64FX (48+4
núcleos)
32 GB memoria HBM2



Rack
192 servidores x 2 cpus
384 CPUs

FUJITSU CONFIDENTIAL

Barcelona Supercomputing Center: Marenostrum IV



MareNostrum 5 - Supercomputador pre-exascale



- **314 Petaflops** peak performance
- Avances científicos como la medicina de alta precisión
- Inversión de 207M€ (MN4 fue de 34M€)

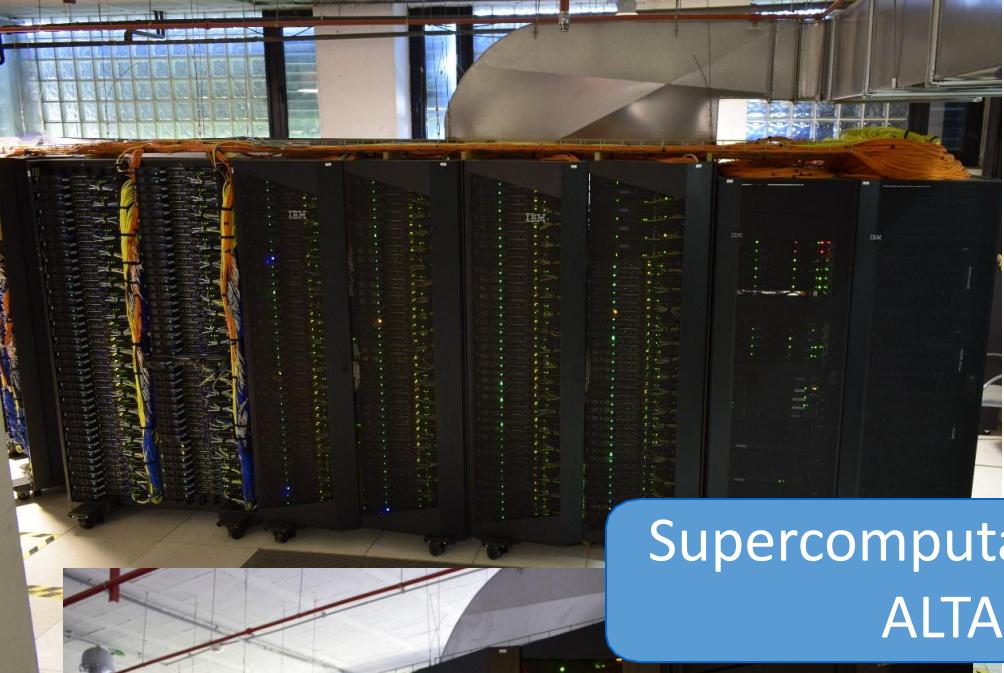


**Barcelona
Supercomputing
Center**

Centro Nacional de Supercomputación

MareNostrum 5 - Supercomputador pre-exascale





Supercomputador cántabro: ALTAMIRA

2012

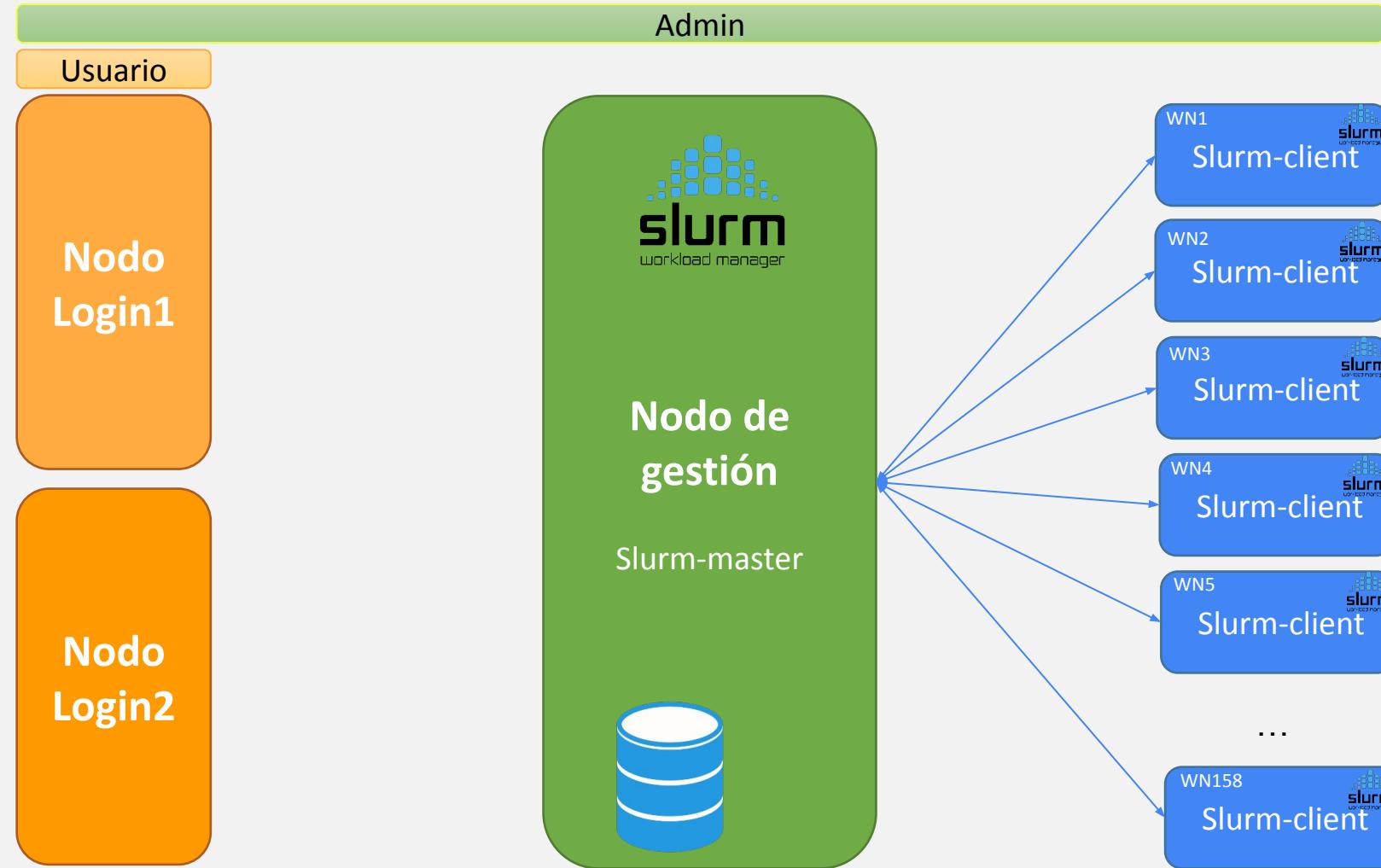
TOP

358

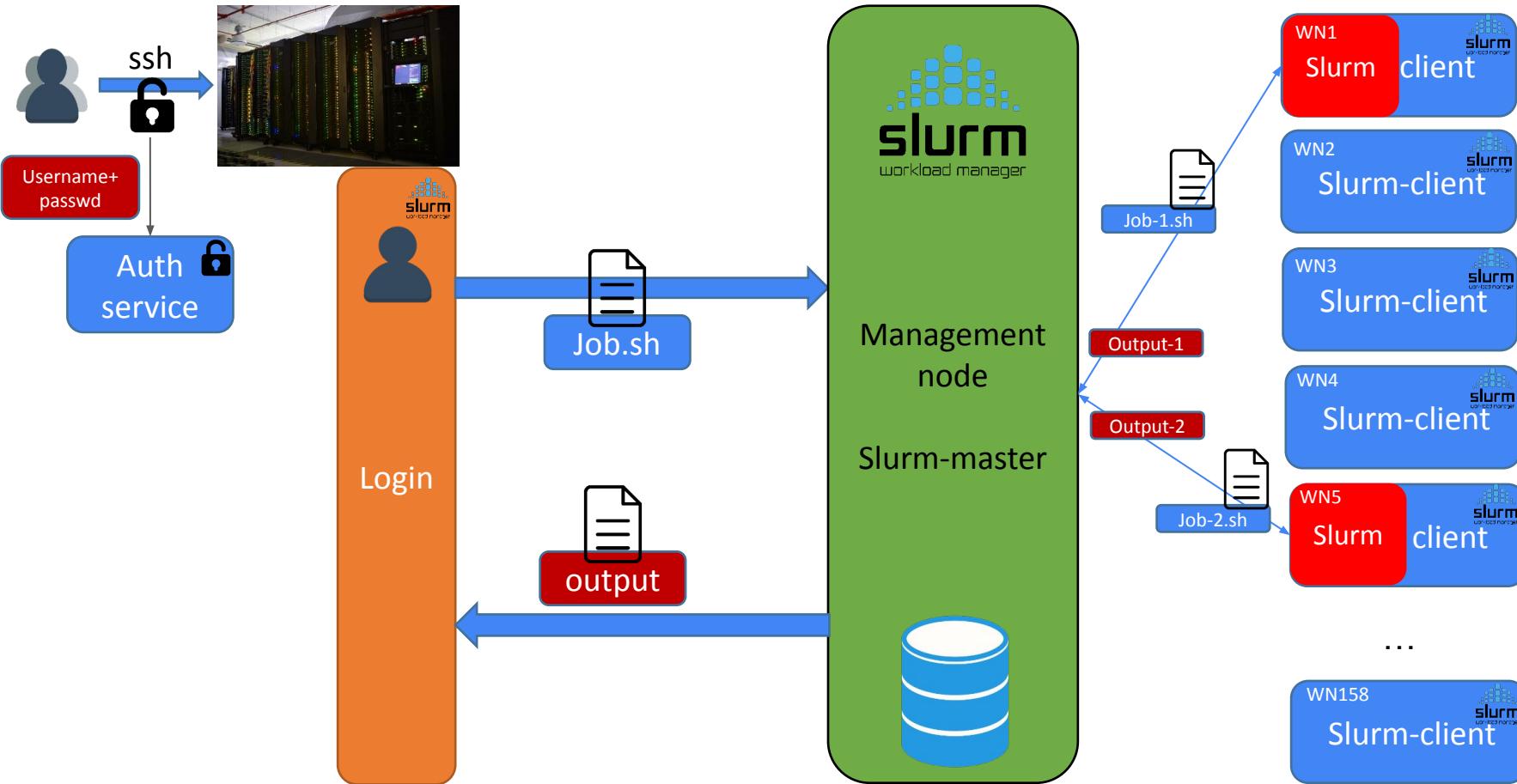
The Lin



Slurm en ALTAMIRA



Ejemplo ejecución en Supercomputador

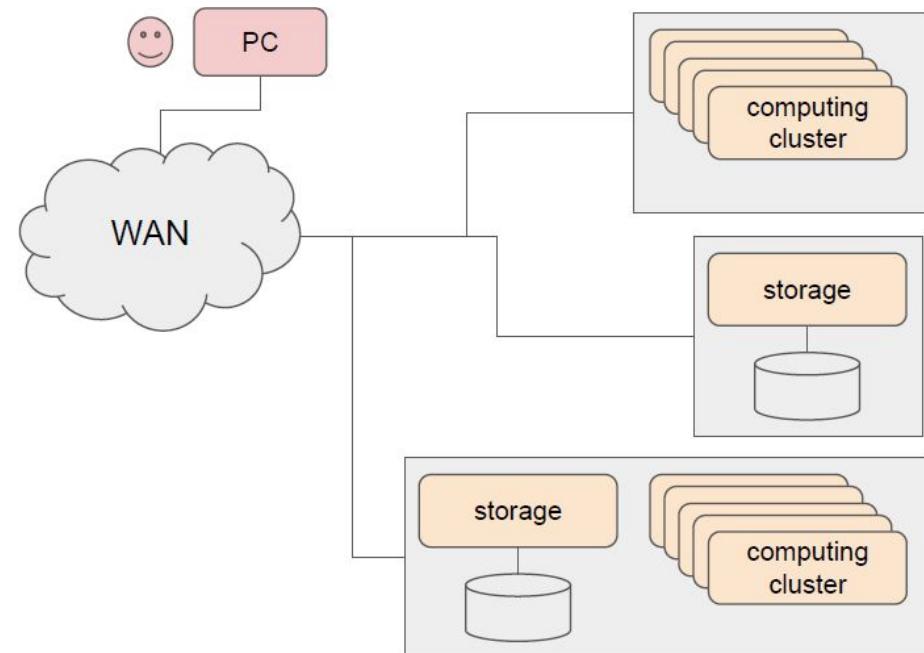


Tipos de clúster de computación

- High Performance Computing (HPC)
- **Grid Computing**
- Cloud Computing

Grid computing (Computación en malla)

- Acceso a recursos de computación de forma distribuida y a nivel global
 - Recursos heterogéneos
 - Distribuidos geográficamente
- Consiste (básicamente) en dar acceso a clústers de computación HTC o HPC y almacenamiento



Grid computing: Aplicaciones

- World Community Grid:
 - Investigación de enfermedades
 - Desastres naturales
 - Problemas medioambientales
- The Worldwide LHC Computing Grid (2002)
 - Guardar, distribuir y analizar 15 PetaBytes de datos generados cada año por el Large Hadron Collider (LHC)
 - Miles de computadores y sistemas de almacenamiento en 170 centros de 41 países
 - Engloba European Grid Infrastructure en Europa, y Open Science Grid in EEUU



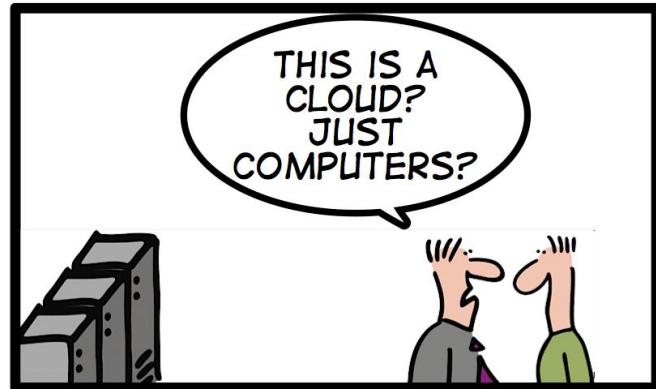
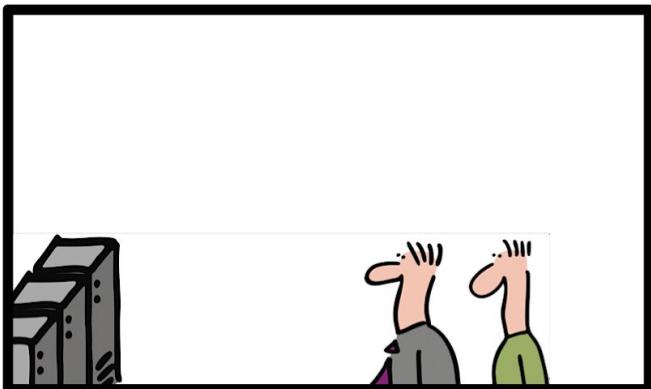
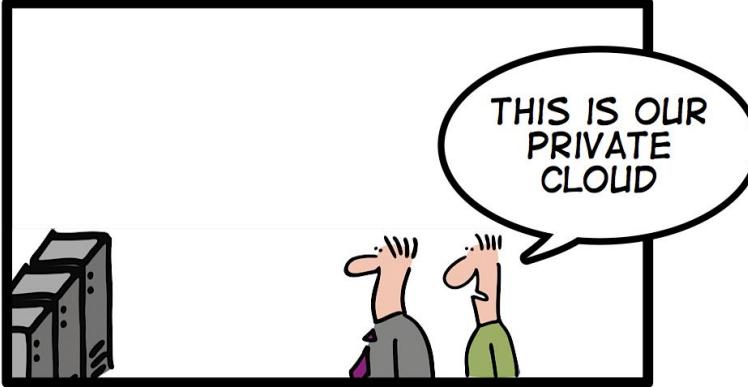
Tipos de clúster de computación

- High Performance Computing (HPC)
- Grid Computing
- **Cloud Computing**

Cloud computing

HOW TO DISILLUSION YOUR BOSS

geek & poke



Cloud computing

- Modelo de provisión de recursos (servidores, almacenamiento, plataformas, aplicaciones) basado en:
 - Provisión flexible, escalable y bajo demanda
 - Acceso a través de red
 - Pago por uso
- Grandes centros de datos con componentes estándar
- Ejemplos:
 - Infraestructura: Amazon Web Services (AWS), Google Cloud Engine, Microsoft Azure, Openstack IFCA

¿Qué no es Cloud computing?

- No es una moda sino un modelo establecido
- No es cualquier solución o servicio disponible en internet
- No es la solución para todo
- No es *outsourcing* de recursos
- No es tan solo acceso remoto a recursos



Cloud Empresarial

Servidor dedicado en la nube

Trabaje desde cualquier lugar con su escritorio Windows y olvídense de los costes de infraestructura informática. Descubra nuestros servidores alojados en un centro de datos de altas prestaciones y mantenido por técnicos cualificados.

- Movilidad Empresarial
- Copias de seguridad de 14 días de la máquina completa
- Ancho de banda ilimitado
- Servicios de Escritorio Remoto incluidos
- Posibilidad de Acceso mediante VPN (Red Privada Virtual)

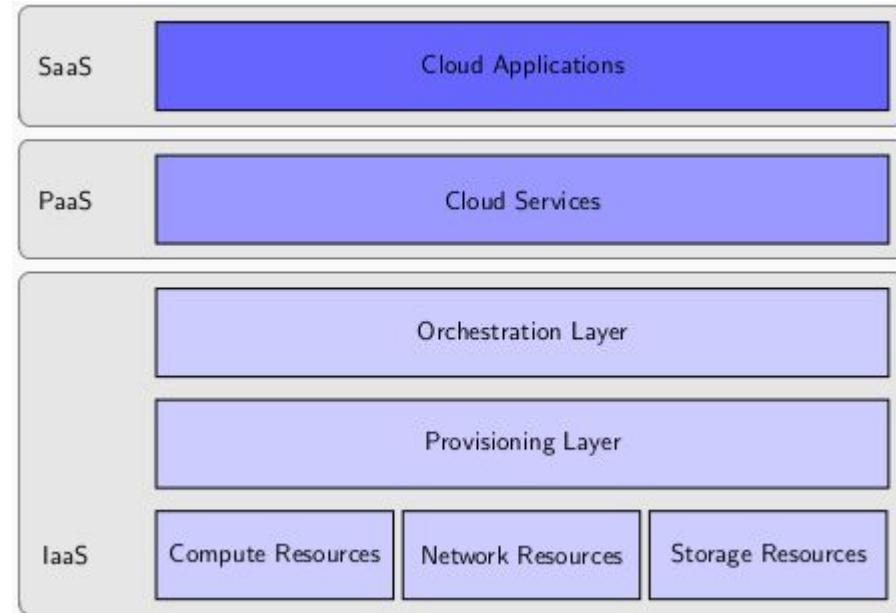
Servicios cloud empresarial



Cloud Computing: Modelos de servicio

Modelo XaaS: Diferentes tipos de Cloud, en función de:

- **Qué** se ofrece bajo demanda
- **A quién** va dirigido.
- Hay que tener en cuenta que los diferentes modelos de Cloud tienen diferentes audiencias (con diferentes conocimientos)



Software as a Service (SaaS)

- El usuario (end-user, customer) accede a aplicaciones ejecutadas sobre todo el stack cloud
- Aplicaciones a las que se accede a través de un cliente ligero (p.e. navegador)
- Ejemplos: Facebook, Google Docs, Dropbox, Spotify, Netflix

Platform as a Service (PaaS)

- Permite ejecutar aplicaciones sin necesidad de manejar la infraestructura
- El usuario (developer) utiliza los recursos (provenientes de un IaaS o no), a través de las herramientas o librerías que el proveedor le proporciona.
- Despliegue transparente, herramientas de gestión, monitorización, etc.
- No tiene control directo de los recursos.
 - El proveedor PaaS se encarga de gestionar los recursos de forma eficiente.
 - Gestión de recursos de computación.
 - Almacenamiento de ficheros.
 - Bases de datos.
 - Balanceo de carga, redundancia.
 - Autenticación y Autorización
 - ...
 - El usuario podrá influir en como se gestionan los recursos.
 - Escalabilidad up/down según carga de trabajos.
 - Establecimiento de límites.
 - ...
- Ejemplos: Windows Azure, Google App Engine, Heroku, etc.

Infrastructure as a Service (IaaS)

- Modelo a más bajo nivel.
- El usuario (sysadmin) obtiene directamente recursos de infraestructura:
 - Computación, red, almacenamiento, etc.
 - Datacenter como servicio.
- El proveedor ofrecerá más o menos abstracción (p.e. configuración de red), pero el usuario es el encargado de gestionar esos recursos:
 - Sistema operativo, aplicaciones, etc.
- Ejemplos: Rackspace, Amazon AWS, OpenStack, etc.

Cualquier recurso como servicio

StaaS Storage as a Service

NaaS Network as a Service

DNSaaS DNS as a Service

XaaS ...

Cloud Computing: Modelos de despliegue

- **Cloud Público**
 - Abierto a todo el mundo
 - Clouds comerciales, operados por terceras personas
- **Community Cloud**
 - Acceso libre a una comunidad (p.e. colaboración científica)
 - Puede estar operada por un tercero
- **Cloud Privado**
 - Operada **para** una organización
 - No tiene porque estar operada **por** la organización
 - No tiene porque ser **propio** de la organización
- **Cloud Híbrido**
 - combinación de varios (p.e. cloud privado que da servicio a una comunidad)

Cloud Computing: Modelos de despliegue II

- Cloud Federado
 - Varios proveedores Cloud operando de forma cooperativa según ciertos estándar acordados:
 - Autenticación,
 - Autorización,
 - APIs,
 - *accounting*,
 - etc.
- Diferentes niveles de federación (tight federation, loose federation)
- Común en colaboraciones científicas (p.e. EGI federated Cloud)



Cloud Computing: Ventajas I

- **Reducción de costes**
 - No hay inversión inicial
 - modelo por uso, posibilidad de acceder a recursos a bajo coste
 - No hay necesidad de una inversión continua
 - Cuando no se necesita la infraestructura no se paga por ella
 - Convertir gasto fijo en variable
- Mayor **adaptabilidad** de la infraestructura
 - Reacción ante picos de demanda
 - Despliegue de infraestructuras de test
- Mayor **flexibilidad** en la infraestructura
 - Reducción de tiempo de despliegue
 - Posibilidad de desplegar el entorno (SO, librerías) que el usuario quiera

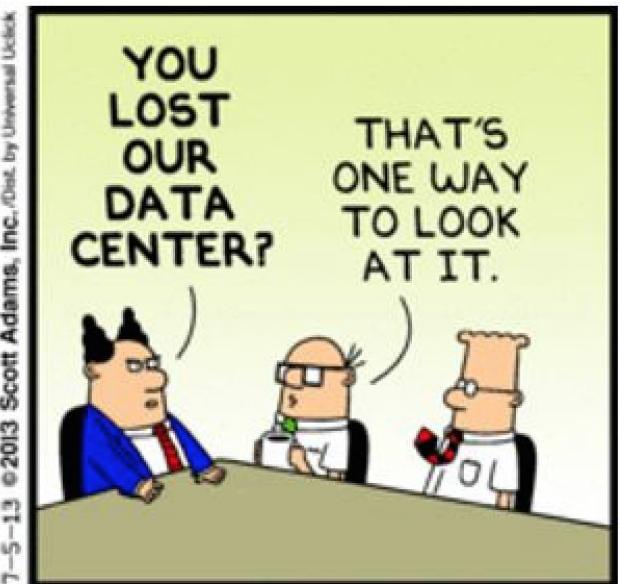
Cloud Computing: Ventajas II

- **Maximizar recursos** en áreas estratégicas
 - Asignar recursos a áreas propias de actividad, no a infraestructuras IT
 - Reducción de inversión inicial
- **Minimizar riesgos**
 - No es necesario personal especializados en la gestión física de la infraestructura
 - Ajusto de recursos a demanda y necesidades

Cloud Computing: Inconvenientes

- **Falta de interoperabilidad y portabilidad**
 - No es posible migrar una infraestructura a otra
 - Dificultad de integrar aplicaciones
- Recursos **fuera de nuestro control**
- **Seguridad:** Entorno compartido, falta de privacidad
- **Fiabilidad:** Dependencia de internet
- Gestión del servicio, pérdida de personal, know-how
- **Variabilidad:** Cuellos de botella, variaciones de rendimiento

Cloud Computing: Inconvenientes



Refrigeración en un clúster de computación

La importancia de refrigerar un supercomputador

- Cuanta más velocidad tenga un procesador más es el calor que produce.
 - **+ velocidad = + calor**
- Un supercomputador puede consumir tanta potencia como una ciudad pequeña.
 - Por tanto, sin un sistema de refrigeración las máquinas no podrían operar adecuadamente e incluso llegar a estropearse
- El **objetivo** de un sistema de refrigeración es mantener los servidores a la temperatura más eficiente posible
- De la misma manera, el clúster es ineficiente si hace frío

Tipos de sistemas de refrigeración

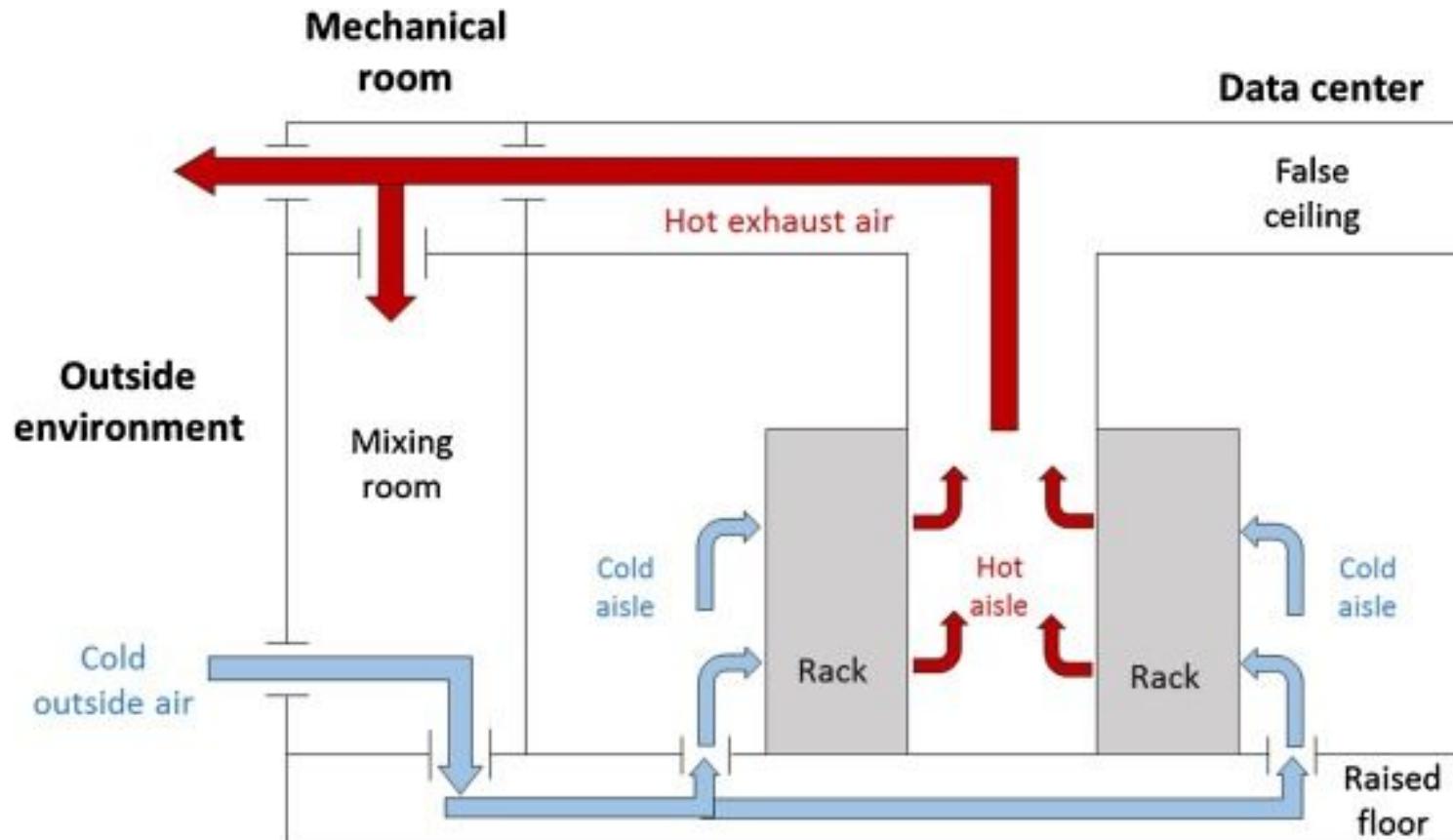
- Las soluciones de refrigeración más comunes son:
 - **Sistema air-cooling:** Método de disipación de calor a través de ventiladores, por ejemplo
 - Puede requerir mucha energía para mover el aire frío a través del servidor
 - **Water cooling system:** Usar tuberías de agua para eliminar el calor de los servidores
 - Hoy en día, **liquid immersion cooling** es un sistema muy innovador
 - Los servidores se sumergen en un líquido refrigerante térmicamente conductor
 - El calor se disipa haciendo circular líquido en contacto directo con los componentes
 - **Free cooling:** utilizar las bajas temperaturas del exterior para ayudar a enfriar la temperatura del clúster



- Se estima que en un CPD el 30 % de la electricidad se destina a los sistemas de refrigeración
- Recuerda que es el equipo el que necesita refrigeración



Free Cooling



Backup

BackUp

Copia de seguridad de los datos originales para poder recuperarlos en caso de pérdida.

- Restaurar ante fallos de sistema, catástrofes naturales o ataques informáticos
- Restaurar pequeña cantidad de archivos que se han eliminado accidentalmente o se han corrompido
- Guardar información histórica para liberar espacio en un disco local
- Redundancia en otros lugares geográficos



Mantenimiento y costes de un CPD

- Equipamiento crítico (entre 10-30% del valor anual):
 - SAIs : 2 SAIs 30KVAs 50000€ (mantenimiento 5000 €)
 - Red: 3 SW + Router 65000€ (mantenimiento 12.000€)
 - Sistemas de refrigeración: 2 equipos 80.000€ (mantenimiento 6.000€)
 - Sistemas de extinción de incendios
 - Sistema almacenamiento: ~2PB 250.000€ (mantenimiento 30.000€)
 - Otros servicios generales: 50.000€
- Personal:
 - 1 E2 - 18k€
 - 1 M2 - 26K€
 - 1 M3 - 31K€
- Nodos de cómputo
 - 50 nodos HPC (incluyendo switches, cableado, fuentes de alimentación, etc.):
~500.000€
 - Deberían ser reemplazados cada 3 - 5 - 7 años (dependiendo de los fondos)
- Electricidad
 - 100Kwh ~ 15.000€ mes ~ 180K€ año

Infraestructuras de computación distribuida

¿Preguntas?

Introducción a los datos masivos y a la Ciencia en abierto
Máster Ciencia de Datos
Aida Palacio Hoz (aidaph@ifca.unican.es)



CSIC
CONSEJO SUPERIOR DE INVESTIGACIONES CIENTÍFICAS

