

# Computación numérica aplicada al análisis de datos científicos

---

*Un estudio de las aplicaciones, métodos y herramientas para el análisis y comunicación de datos experimentales.*

Pol Marcos Payà

Modalidad: Tecnológico

Tutor: Daniel Parcerisas

Fecha: 08-01-2019

Centro: Sagrada Familia



# Índice

• Introducción y objetivos	.....	5
• Parte teórica	.....	8
○ Análisis numérico	.....	9
○ Análisis numérico aplicado a la ciencia	.....	17
○ Octave	.....	24
• Parte práctica	.....	36
○ Programación de Scripts en Octave	.....	38
○ Barcelona Supercomputing Center	.....	45
○ Entrevista a José Luis Vázquez-Poletti	.....	52
• Conclusiones	.....	56
• Referencias	.....	59
• Anexos	.....	64
○ Análisis del software PixetPRO	.....	65
○ Práctica Detección de partículas	.....	72
○ Práctica en la UB	.....	82
○ Creación de un servidor de Wordpress	.....	84
○ Creación de una página web	.....	93
○ Parte teórica común.	.....	98



# 1. Introducción

La computación, actualmente, es el eje fundamental de la comunicación y desarrollo en el ámbito de las telecomunicaciones. Todo está ligado a ella y forma parte de cualquier proyecto tecnológico y científico.

Aparte de su importancia para el mundo científico-tecnológico, este es un tema que me apasiona ya que en un futuro muy próximo quiero estudiar una ingeniería relacionada con las telecomunicaciones.

Este trabajo forma parte de un proyecto global que empezó de manera conjunta con otros compañeros a partir de unas prácticas con un detector de partículas que nos cedió el doctor Rafael Ballabriga

El trabajo está estructurado en dos grandes bloques: parte práctica y parte teórica, así como un anexo.

En la parte teórica me voy a centrar en el desarrollo del análisis numérico desde su origen hasta la actualidad, así como de los diferentes proyectos que lo utilizan.

Primero empezaré haciendo una introducción sobre el análisis numérico: qué es, qué métodos utiliza en las diferentes ecuaciones, los errores y estabilidad numérica.

A continuación trataré de su aplicación en el mundo científico- tecnológico y su evolución en este ámbito

En tercer lugar me centraré en los diferentes proyectos que utilizan el lenguaje numérico como herramienta básica de trabajo. Haré una comparativa entre cloud computing y los diferentes tipos de nube y grid explicando su importancia dentro de cada uno de ellos y como el primero ha absorbido prácticamente al segundo en el mundo científico tecnológico.

Y para terminar me centraré en Octave que ha sido el eje vertebrador del trabajo. Hablaré de su estructura, usos y funcionamiento.

En esta parte práctica he combinado el trabajo personal en casa y con el ordenador con una serie de visitas y/o conferencias/entrevistas, unas presenciales como en el caso de la visita al superordenador y otras on-line como la entrevista con el doctor José Luís Vázquez-Poletti.

La parte práctica del trabajo se centrará en 4 puntos básicos que nos servirán para entender todo el proceso.

Empezaré con el análisis detallado de una serie de scripts explicando cada uno de los pasos dados y lo acompañaré de unos ejemplos que servirán de guía y soporte visual.

Continuaré explicando la visita a Barcelona Supercomputing Center y de los usos del superordenador dentro del campo de la investigación.

También detallaré la información extraída de la entrevista al doctor Vázquez-Poletti, ingeniero electrónico y doctor en la Universidad Complutense de Madrid.

Por último, en el apartado de los anexos, se ha incorporado mi contribución a la parte teórica de los cuatro trabajos relacionados con el TimePix. También se ha añadido en esta sección la parte teórica realizada por mis compañeros. A continuación explicaré cómo creé un servidor de Wordpress y su instalación para la creación de la página web de comunicación de las actividades realizadas con el detector.

Terminaré con el informe de la práctica que hicimos en el colegio para aprender a usar el detector TimePix y la visita al laboratorio de física moderna de la universidad de Barcelona en la que realizamos una práctica usando un generador de rayos X.

## Objetivos

Al principio del trabajo me marqué una serie de objetivos que fueron:

- Aprender a usar un detector TimePix para obtener medidas de diferentes tipos de radiación en diferentes experimentos.
- Usar el software PixetPRO para configurar el detector y obtener los datos.
- Aprender cómo las partículas y los rayos X viajan a través de materiales mediante varios experimentos.
- Aprender a utilizar herramientas para crear una página web y aprender a gestionarla.
- Instalar y configurar un servidor local de APACHE con PHP y una base de datos mySQL.
- Instalar un sistema de gestión de contenidos (WORDPRESS) sobre el servidor local
- Crear una página web para comunicar los resultados de las actividades realizadas con el detector y administrarla de manera adecuada.
- Aprender cual es la estructura y el funcionamiento de un programa de análisis numérico (OCTAVE)
- Aprender la sintaxis del lenguaje de programación de Octave para realizar pequeños scripts de algoritmos de análisis numéricos.

- Profundizar en el funcionamiento de PixetPRO haciendo una entrevista con algunas de las personas que se dedican a programarlo.
- Investigar qué soluciones se han encontrado para analizar datos en proyectos científicos con gran análisis de datos (genoma humano, síntesis de drogas, física de cuántica, exploración espacial, etc.)
- Comprender la importancia de la tecnología de la información y el análisis de los datos en el desarrollo científico actual, entrevistando un soporte informático en diferentes proyectos científicos
- Investigar la organización y el sistema de trabajo de proyectos científicos que utilicen supercomputadores para realizar investigaciones.
- Saber que es el análisis numérico, cuáles son sus partes y características (ecuaciones, métodos....) y su uso en ciencia e investigación.
- Conocer qué tipos de almacenamiento y organizaciones de trabajo se usan en computación aplicada a análisis de datos científicos (cloud computing,grid.....)
- Descubrir las limitaciones impuestas por el error en los tratamientos numéricos aproximados y como tratar con dichos errores.
- Investigar que es la estabilidad numérica para así poder comprender su importancia en el análisis numérico.

# PARTE TEÓRICA

---

## 2. Análisis numérico

Este apartado trata sobre qué es el análisis numérico. Tras una breve explicación introductoria y la definición, se presentarán una serie de técnicas de análisis numérico agrupadas por tipos.

El análisis numérico es el estudio de algoritmos para los problemas de análisis matemático y se usan para dar soluciones muy aproximadas a determinadas situaciones que plantean problemas difíciles de resolver.

En matemáticas, lógica, ciencias de la computación y disciplinas relacionadas, un algoritmo es un conjunto prescrito de instrucciones o reglas bien definidas, ordenadas y finitas que permiten llevar a cabo una actividad mediante pasos sucesivos que no generen dudas a quien deba hacer dicha actividad.

Dados un estado inicial y una entrada, siguiendo los pasos sucesivos se llega a un estado final y se obtiene una solución.

El análisis numérico se usa en una gran variedad de casos como para predecir numéricamente el tiempo que va a hacer (tendría que estar acompañado por un trabajo de computación) o para predecir la trayectoria de una nave espacial de manera muy precisa.

El análisis, tradicionalmente, se había usado para labores de ingeniería o física pero a partir del siglo XXI, se empezó a usar en campos como la biología y medicina.

### 1.1. Ecuaciones y cálculos

La herramienta fundamental del análisis numérico es la ecuación. En este apartado veremos los diferentes tipos de ecuaciones y los distintos usos que tienen en el ámbito científico-tecnológico.

- **Las ecuaciones diferenciales estocásticas:** Fueron usadas por primera vez en física para describir el movimiento Browniano en los *Annus Mirabilis papers*, escritos por Albert Einstein y base de la física moderna. A partir de estas ecuaciones se produjo un cambio de visión del espacio, la masa, el tiempo y la energía. Otros investigadores las usaron primero pero para mí, fue Einstein quien les dio la importancia que actualmente tienen como pioneras de los pilares de la física moderna.

Son ecuaciones diferenciales en las que, por lo menos una de ellas, es un proceso estocástico, es decir, un objeto matemático definido como una colección de variables aleatorias. Se pueden dividir y/o utilizar en:

- **El cálculo estocástico de Itô**, que se emplea para generalizar el método de Euler de pasar ecuaciones diferenciales ordinarias a ecuaciones diferenciales estocásticas.

-**El cálculo estocástico de Stratonovich**: se puede usar la regla de la cadena para resolverse lo que hace que sea más sencillo definir las variables diferenciales (es un tipo de variable más o menos similar al espacio lineal que permite hacer cálculos) . En cambio el cálculo de la regla de la cadena con el cálculo de Itô es más complicado cuando tienes que analizar estas variables.

Esta es la integral:

$$\sum_{i=0}^{k-1} \frac{X_{t_{i+1}} + X_{t_i}}{2} (W_{t_{i+1}} - W_{t_i})$$

-**La ecuación de Langevin** esta ecuación es empleada en los campos de la física ya que es una forma de resolver los problemas más general.

Si nos basamos en el movimiento Browniano nos describe un movimiento de partículas aleatorio en un fluido o gas donde actuarán diferentes variables.

$$m \frac{d^2 x}{dt^2} = -\lambda \frac{dx}{dt} + n(t)$$

X=Posición de la partícula

m=Masa de la partícula

n(t)=el término del sonido

## 1.2. Métodos directos e indirectos

Los métodos **directos** son aquellos que dan una respuesta precisa. Se pueden incluir en este tipo de métodos:

-La **eliminación Gaussiana** que es un algoritmo usado para resolver sistemas de ecuaciones lineales, para encontrar el rango de una matriz y para calcular la determinante de la matriz o su matriz cuadrada invertible.

Para utilizar este método en las matrices, que es lo que nos interesa a nosotros, debemos usar una serie de operaciones para modificar la matriz y hacer que la esquina inferior de la derecha esté llena de ceros y por tanto se transforme en una matriz triangular. Esto se puede hacer con la ayuda de tres operaciones:

1. Intercambiando dos filas.
2. Multiplicando una fila por un número distinto a cero
3. Añadiendo un múltiplo de una fila a otra fila

Con esto conseguiremos que la matriz quede en forma de escalón reducida, es decir, cada columna que contiene un coeficiente tendrá un cero en una de sus partes.

Este ejemplo nos enseña como las matrices del final son las que tienen la forma escalonada, a pesar que, en todas se han realizado operaciones elementales.

$$\left[ \begin{array}{ccc|c} 1 & 3 & 1 & 9 \\ 1 & 1 & -1 & 1 \\ 3 & 11 & 5 & 35 \end{array} \right] \rightarrow \left[ \begin{array}{ccc|c} 1 & 3 & 1 & 9 \\ 0 & -2 & -2 & -8 \\ 0 & 2 & 2 & 8 \end{array} \right] \rightarrow \left[ \begin{array}{ccc|c} 1 & 3 & 1 & 9 \\ 0 & -2 & -2 & -8 \\ 0 & 0 & 0 & 0 \end{array} \right]$$

-La **descomposición QR** se basa en la descomposición de una matriz A en la cual A=QR y (Q) es una matriz ortogonal y (R) es una matriz superior triangular.

Hay diferentes usos pero yo me centraré en la descomposición QR utilizado en la computación y aunque se pueden usar 3 tipos de procesos me basaré en el Proceso de Gram-Schmidt.

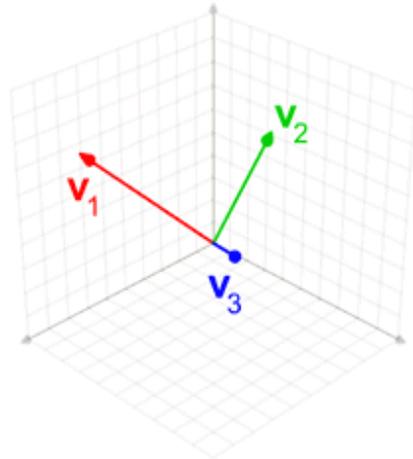


Imagen 1: Descomposición QR de 3 vectores

1. Proyectaremos un vector en el plano donde  $(u, v)$  serán el producto interior de los vectores  $(u, v) = u \cdot v$ .
2. Se proyectara el vector  $v$  ortogonalmente en la línea distribuida del espacio del vector  $u$ .
3. Si el vector  $u$  es igual a 0 definiremos una proyección donde  $v=0$ .

Un ejemplo práctico es el **espacio euclídeo** :

Partiremos de esta base

$$S = \left\{ \mathbf{v}_1 = \begin{pmatrix} 3 \\ 1 \end{pmatrix}, \mathbf{v}_2 = \begin{pmatrix} 2 \\ 2 \end{pmatrix} \right\}.$$

Ahora utilizaremos esta técnica para obtener los vectores ortogonales:

$$\mathbf{u}_1 = \mathbf{v}_1 = \begin{pmatrix} 3 \\ 1 \end{pmatrix}$$

$$\mathbf{u}_2 = \mathbf{v}_2 - \text{proj}_{\mathbf{u}_1}(\mathbf{v}_2) = \begin{pmatrix} 2 \\ 2 \end{pmatrix} - \text{proj}_{\begin{pmatrix} 3 \\ 1 \end{pmatrix}} \left( \begin{pmatrix} 2 \\ 2 \end{pmatrix} \right) = \begin{pmatrix} 2 \\ 2 \end{pmatrix} - \frac{4}{5} \begin{pmatrix} 3 \\ 1 \end{pmatrix} = \begin{pmatrix} -2/5 \\ 6/5 \end{pmatrix}.$$

Después comprobaremos que los vectores  $u_1$  y  $u_2$  son ortogonales. Serán ortogonales si el producto de estos cálculos resulta cero:

$$\langle \mathbf{u}_1, \mathbf{u}_2 \rangle = \left\langle \begin{pmatrix} 3 \\ 1 \end{pmatrix}, \begin{pmatrix} -2/5 \\ 6/5 \end{pmatrix} \right\rangle = -\frac{6}{5} + \frac{6}{5} = 0,$$

Si no nos da cero podemos “normalizarlos” como se muestra a continuación, dividiéndolos.

$$\mathbf{e}_1 = \frac{1}{\sqrt{10}} \begin{pmatrix} 3 \\ 1 \end{pmatrix}$$

$$\mathbf{e}_2 = \frac{1}{\sqrt{\frac{40}{25}}} \begin{pmatrix} -2/5 \\ 6/5 \end{pmatrix} = \frac{1}{\sqrt{10}} \begin{pmatrix} -1 \\ 3 \end{pmatrix}.$$

Con los **métodos iterativos** no se acaba con unos pasos determinados de aquí que sean usados en problemas que tienen gran número de variables. La resolución de estos problemas sería prácticamente imposible si utilizáramos el método directo ya que nos consumiría una gran cantidad de recursos.

Dentro de estos métodos para resolver sistemas de ecuaciones lineales se pueden destacar dos:

- El **método de los iterativos fijos**: Se basa en la resolución del sistema usando un **operador** que funciona de manera que, mapea las soluciones en un espacio determinado, para producir otros elementos en el mismo espacio. Si hablamos de temas de computación usaremos operadores lineales.

-

Los operadores **lineales** en computación son usados para operar con operadores diferenciales, los cuales son prácticos ya que te permitirán tomar operaciones abstractas y te devolverán otra función.

Su estructura en computación y con el lenguaje de computación de Octave sería esta:

```

#include <iostream>

auto twice = [](auto f, int v)
{
    return f(f(v));
};

auto f = [](int i)
{
    return i + 3;
};

int main()
{
    std::cout << twice(f, 7) << std::endl;
}

```

-Los métodos del subespacio de Krylov, se usan para resolver sistemas de ecuaciones lineales utilizando matrices dispersas. Se basan en un conjunto de métodos como por ejemplo :

a) *El método del gradiente conjugado* usado para resolver sistemas de ecuaciones lineales pero solo se podrán resolver sistemas cuyas matrices sean simétricas y definidas positivas.

El código de computación utilizando este último método es:

```

function [x] = conjgrad(A,b,x0)

    r = b - A*x0;
    w = -r;
    z = A*w;
    a = (r'*w)/(w'*z);
    x = x0 + a*w;
    B = 0.783564;

    for i = 1:size(A,1);
        r = r - a*z;
        if( norm(r) < 1e-10 )
            break;
        end if
        B = (r'*z)/(w'*z);
        w = -r + B*w;
        z = A*w;
        a = (r'*w)/(w'*z);
        x = x + a*w;
    end

```

### 1.3. Errores

En todo proceso de análisis numérico para la resolución de un problema usando un algoritmo siempre se producen errores, bien por el propio algoritmo, por las limitaciones o aproximaciones realizadas o por otros factores.

En el análisis numérico lo que se pretende es estimar los errores de cálculo y después encontrarles solución.

Algunos de los errores más comunes son:

**-Error de redondeo:** Este es un tipo de error que aparece cuando usamos ecuaciones de aproximación. Aunque cuando más aparecerán será cuando utilicemos números finitos para representar cifras que, en realidad, tienen muchos números más o simplemente son infinitos.

Aquí se pueden ver algunos ejemplos de errores en representaciones por muy pequeños que sean.

Notation	Representation	Approximation	Error
$1/7$	$0.\overline{142857}$	0.142857	0.000000 $\overline{142857}$
$\ln 2$	0.69314718055994530941...	0.693147	0.00000018055994530941...
$\log_{10} 2$	0.30102999566398119521...	0.3010	0.00002999566398119521...
$\sqrt[3]{2}$	1.25992104989487316476...	1.25992	0.00000104989487316476...
$\sqrt{2}$	1.41421356237309504880...	1.41421	0.00000356237309504880...
$e$	2.71828182845904523536...	2.718281828459045	0.00000000000000023536...
$\pi$	3.14159265358979323846...	3.141592653589793	0.00000000000000023846...

Imagen 2: Tabla comparativa de diferentes errores típicos

**-Errores de truncamiento:** Hay dos tipos dependiendo de si está causado por una repetición o por varias repeticiones.

*Los errores de truncamiento locales:* Son aquellos causados ya que en el incremento de nuestra función se asume un error por cada repetición.

*Los errores de truncamiento globales :* Es aquel tipo de error que sucede por culpa de la acumulación de los *errores de truncamiento locales* en cada repetición.

Esto nos permitirá calcular este tipo de errores solamente con establecer un límite superior en los *errores de truncamiento locales*, aunque solo podremos hacerlo si existe un incremento de función relativamente grande.

## 1.4. Estabilidad numérica

La estabilidad numérica es una propiedad que se da generalmente en los algoritmos numéricos.

Si un problema está en '**buenas condiciones**', quiere decir que la solución cambiará mínimamente a lo largo de varios cálculos y por tanto su error será mínimo (todos los algoritmos tendrán un cálculo aunque sea muy pequeño), en cambio si el problema está en '**malas condiciones**' el error que tiene ese algoritmo es tan grande que modifica el resultado de las operaciones al largo de varios cálculos, cosa que no nos interesa para nada y debemos arreglarlo.

Dentro de la estabilidad existen dos tipos dependiendo de si lo usamos para álgebra lineal o si lo usamos para ecuaciones diferenciales.

En la estabilidad numérica para el álgebra **lineal** existen diferentes definiciones aunque la más típica es la de **estabilidad mixta** que combina dos de los errores más característicos, el *error de avance* y el *error de retroceso*.

*El error de avance* es la diferencia entre el resultado y la solución, en cambio *el error de retroceso* es aquel que nos dice el problema que tiene el algoritmo cuando está resuelto.

En otras palabras, un algoritmo será estable mediante el método mixto cuando resuelva un problema aproximadamente.

En la estabilidad numérica en **ecuaciones diferenciales** existirán diferentes conceptos de estabilidad dependiendo del uso de cada uno.

Si hablamos de ecuaciones diferenciales ordinarias tenemos varios métodos: si estamos hablando, por ejemplo, de sistemas dinámicos utilizaremos un sistema de estabilidad llamado método de Lyapunov. En cambio, si estamos resolviendo una ecuación 'rígida' siempre deberemos utilizar métodos muy estables.

En cambio si estamos hablando de ecuaciones diferenciales parciales, una ecuación será solamente estable si la variación total de la solución en un tiempo determinado permanece, bajando paso a paso, hacia el cero. Para este caso utilizaremos el **teorema de equivalencia de Lax** o el método de **Von Neumann** aplicado en programas de diferencia finita utilizado en este tipo de ecuaciones.

Hay veces que la estabilidad numérica se logra mediante *la difusión numérica*.

La difusión numérica es un término que se utiliza en el redondeo de las soluciones juntando otros errores sin sumarlos. Esto conlleva a que el método de computación falle a causa de estos errores.

## 3. Análisis numérico aplicado a la ciencia

A continuación os haré una introducción al análisis numérico aplicado a la ciencia.

Hablaré sobre su evolución, los sistemas que hay en la actualidad y algunos proyectos de análisis numérico aplicados a la ciencia.

### 1.5. Historia

Fue usado por primera vez por los babilonios (2000 a.c) para componer tablas matemáticas ( se encontró una tablilla de barro con los cuadrados enteros del 1 al 30).

Arquímedes lo utilizó en los polígonos regulares para aproximar el área del círculo.

Se sabe que el análisis numérico fue usado para la preparación de tablas astronómicas hasta el siglo XVII. En el momento en el que se descubre el álgebra este método decae y se renuevan las ramas de las matemáticas y de manera automática la del análisis numérico.

A partir de ese momento se empezaron a descubrir muchos elementos con los que se mejoró el análisis numérico y nos permitió conocerlo como lo hacemos ahora.

En 1620, fue publicada por Neper la primera tabla de logaritmos, esto permitía por ejemplo resolver operaciones con números muy grandes fácilmente convirtiéndolos en logaritmos y al final volviéndolos a transformar mediante antilogaritmos.

Más tarde Jacob Stirling y Brook Taylor introdujeron en el análisis numérico, el cálculo de diferencias finitas algo que actualmente es muy estudiado en el análisis numérico.

Con la llegada de las máquinas de cálculo automático se estimuló bastante el análisis numérico por el simple hecho que, se podían realizar cálculos más complicados con relativa rapidez y facilidad.

Finalmente nos encontramos en la actualidad. El análisis numérico se realiza con la ayuda de ordenadores pero solo para ejecutar las instrucciones ya que el ordenador es el encargado de realizar todas las tareas de computación.

## 1.6. Proyectos que utilizan el análisis numérico

### 3.1.1. Cloud computing:

Cloud computing es un modelo de provisión de servicios de computación.

Un modelo de provisión es un servicio que nos provee de algo que necesitamos. Por ejemplo, en casa, una serie de servicios de provisión nos proveen de gas, electricidad.... En el caso de la computación en la nube diremos que son servicios de computación.

Sus principales características son:

-Reducción de los costes de los servicios de computación debido a que tú alquilas por horas o meses los servicios de computación dependiendo de lo que necesites para tu proyecto.

Antes, cuando no se usaba la computación en la nube para realizar proyectos privados, se necesitaba bastante financiación por el simple hecho de que habían de comprar toda la infraestructura y después de realizar el proyecto, esa estructura se dejaba de utilizar. En cambio, actualmente, para realizar un proyecto privado solo necesitas contratar ese tiempo de alquiler de las máquinas de computación.

Los precios dependen del tiempo que necesites tener los datos ya que a más tiempo más capacidad de computación y por ende más dinero.

-No necesitas moverte de tu centro de investigación ya que puedes acceder mediante internet al centro de aprovisionamiento y desde allí realizar todos los trabajos de computación que necesites ya que todos los datos están almacenados en la nube después de realizar los trabajos.

-No necesita mantenimiento ya que no dispones de máquinas físicas, las contratas a empresas como Amazon y son ellos los que se encargan de mantenerlas.

-Mejora la productividad pudiendo trabajar de manera cooperativa, es decir, los diferentes usuarios pueden acceder a esos datos desde cualquier lugar.

Se puede decir que es un servicio elástico ya que puedes contratar el volumen de computación que necesites en ese momento. De este modo lo que se consigue es, casi de manera inmediata, aumentar o disminuir el volumen de computación, ahorrando muchos recursos ya que solo contratas lo que vas a usar.

## **Tipos de nube**

- **Nube privada:**

Esta infraestructura es operada solamente por una organización, es decir, los servicios de computación son usados solamente por la empresa.

Este tipo de nube plantea varios problemas de seguridad ya que cada vez se plantean más vulnerabilidades que deben ser resueltas.

Otro de los problemas es que necesitas disponer de un lugar para tener tus máquinas físicas. Esto supone una gran desventaja ya que al que tener que comprar, construir y manejar esta estructura no se aprovecha la gestión práctica como sucede con otros modelos de computación en la nube.

- **Nube pública:**

Este tipo de nube se llama así ya que sus servicios se prestan para el uso público, es decir, sus servicios pueden ser contratados por cualquier usuario que los alquile.

Estos servicios son muy usados por los equipos de investigación ya que les permiten poder satisfacer sus tareas de computación de manera económica debido a que no tienen que invertir en una infraestructura. Otro punto a favor es que pueden realizar su trabajo casi de manera inmediata porque la gestión de estas nubes es muy eficiente (dependiendo de la potencia de máquina que alquiles podrás realizar el trabajo en un tiempo u otro).

Su estructura es la misma que la de la nube pública pero una de las diferencias notables radica en la seguridad. Siempre la nube privada será más segura por el simple hecho que, una red pública está conectada a una red que no es de confianza y por tanto las posibilidades de hacking aumentan exponencialmente.

El acceso a estos tipos de nube se hace a través de servicios de internet .

- **Nube híbrida:**

Es la composición de dos o más nubes las cuales pueden ser privadas, públicas o comunitarias. Estas nubes siguen siendo entidades separadas ofreciendo así los beneficios de los múltiples modelos de implementación y permiten ampliar la capacidad de un solo servicio con otro servicio en la nube por agregación, integración o personalización.

Para este tipo de nube existen diferentes tipos de combinaciones ,algunas de ellas son:

-Puede ser que una empresa almacene datos sensibles de sus clientes en una nube privada, cosa que confiere más seguridad, pero para llevar a cabo la aplicación de computación se utiliza una nube pública.

-Otro caso puede ser que en una empresa se necesite, en un momento dado, un aumento de la capacidad de computación y para satisfacer ese pico optes por una nube pública.

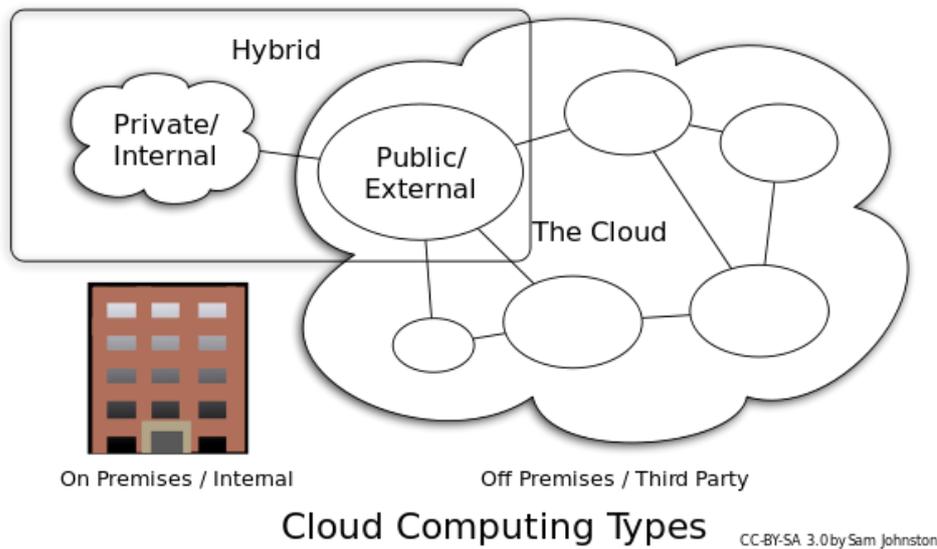


Imagen 3: Diferentes tipos de nube que existen

### 3.1.2. Grid

Este tipo de computación se basa en la utilización de forma coordinada de recursos que no están sujetos a un control centralizado, es decir, que para cubrir toda la potencia de cómputo se utilizan diferentes tipos de máquinas que van desde los superordenadores, a otros con una potencia inferior, sin importar su capacidad individual.

Cada uno de los ordenadores pueden estar administrados por diferentes instituciones o usuarios los cuales pueden colaborar desde cualquier lugar. Y cada una de las máquinas que forman la “malla” están conectadas mediante redes de área extensa, como por ejemplo internet.

Este sistema fue usado la primera vez por el CERN para intercambiar datos con diferentes instituciones y de este modo facilitar la integración de recursos computacionales. Además fue el más usado durante la década de los 90 aunque actualmente se ha visto sustituido por el cloud computing.

Sus características más importantes son:

-La capacidad de alternar entre sistemas. Gracias a eso lo que conseguimos es no tener que calcular cuánta capacidad de computación necesitamos ya que cuando

necesitamos más volumen de computación, ésta se asignará desde la granja de recursos y por tanto no habría necesidad de calcular la capacidad de los sistemas en función de los picos de trabajo.

-La alta disponibilidad ya que si falla un servidor, los servicios que estaban dedicados a este servidor se reasignan a los otros servidores.

-Su bajo coste también es una característica muy importante por el simple hecho de que no debemos disponer de grandes servidores y podemos usar componentes de bajo coste.

A pesar de todas estas ventajas, este sistema también tiene una serie de problemas que deben ser solucionados:

-Uso de recursos heterogéneos. El sistema grid debe poder manejar cualquier tipo de recurso que trabaje en el sistema principal.

-Se han de desarrollar aplicaciones específicas para manejar grid solventando el problema de comunicación lenta y no uniforme.

Es por tanto un sistema muy poco usado ya que lo ha substituido la computación en la nube, mucho más eficiente y práctica.

Para este tipo de computación existen varios proyectos los cuales utilizan tu ordenador para poder llegar a una capacidad superior de computación.

Trabajan de manera que tu donas tu ordenador para que se una a esa "malla".

A continuación voy a enseñaros los proyectos que tienen más éxito.

#### **SETI:**

Los primeros proyectos surgieron patrocinados por la NASA en los años 70, el proyecto más conocido es SETI@Home, apoyado por miles de personas.

Actualmente también existen diferentes proyectos de este tipo. Estos proyectos se encargan de buscar vida extraterrestre inteligente y lo hacen mediante el análisis de señales electromagnéticas capturadas en distintos radiotelescopios o enviando mensajes con distintas frecuencias con la esperanza de que sean contestados.

Hasta ahora ninguno de estos proyectos ha conseguido recibir una señal de vida extraterrestre.

#### **Einstein@Home:**

Este proyecto se basa en buscar púlsares (estrellas de neutrones) usando dos detectores de ondas gravitacionales.

Este programa trabaja primero procesando los datos de los observatorios, utilizando Transformaciones Rápidas de Fourier. Las señales que resultan son analizadas utilizando el método de igualdad mediante filtrado. Con esto se consigue calcular cómo sería la señal si hubiera una fuente de ondas en la parte que se está analizando. Por último, se comparará con la señal calculada y si coinciden se analizará con más profundidad.

El programa se puede ejecutar de dos maneras, de forma continua o en modo salvapantallas (cuando no está siendo usado). Este último mostrará las estrellas descubiertas hasta el momento y la zona donde está apuntando el observatorio.

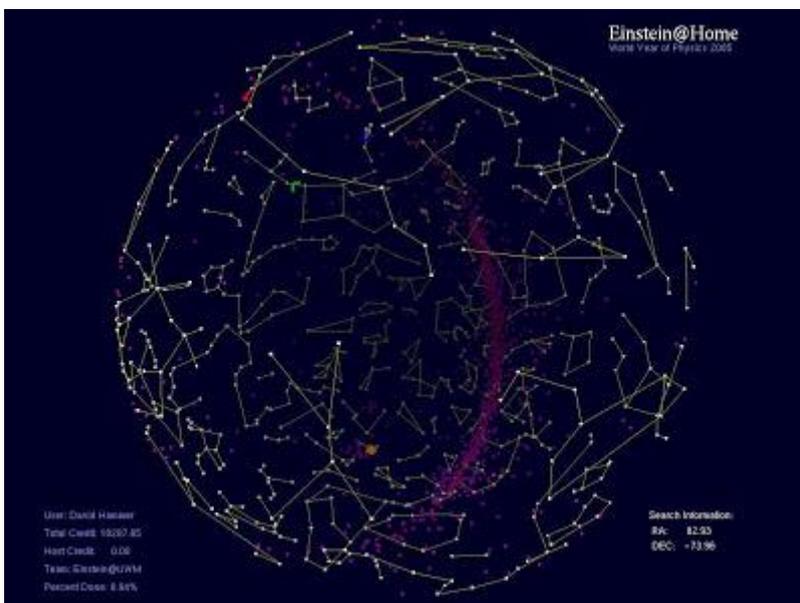


Imagen 4: Fondo de pantalla del proyecto Einstein@Home

**Climateprediction.net:**

Con este proyecto lo que se pretende es recrear mediante la ejecución de millones de modelos, las diferentes variables que existen en el clima.

Estas simulaciones se llevan a cabo durante el tiempo de inactividad de los ordenadores. Este tiempo se dona gratuitamente, dando lugar a lo que se conoce como modelo voluntario de computación.

Este proyecto es ejecutado por la Universidad de Oxford y actualmente es el proyecto que ha aprovechado más potencia informática y en consecuencia ha generado muchos más datos que cualquier proyecto de este tipo

## 4. Octave

Octave es un programa de computación creado en 1988. Originalmente se creó como un apoyo para un libro de texto en el cual se diseñaban reactores químicos. Empezó a ser usado como lo que es hoy en día en 1992 y la primera versión fue lanzada en 1993.

Está diseñado usando un lenguaje de programación estructurado escrito en C++ (un tipo de lenguaje basado en C y en otros programas como UNIX).

Su sintaxis es muy similar a la de Matlab cosa que te permite ejecutar scripts de Matlab en Octave.

También utiliza un **intérprete**, un programa que te permite ejecutar por ejemplo scripts sin necesitar que esté compilado en nuestro ordenador.

Trabaja primero analizando el código y traduciéndolo a un código llamado de representación intermedia que es usado en máquinas virtuales o en los compiladores para representar el código fuente y así también modificándolo. Existe una colección de compiladores producidos por **GNU project** que incluyen varios lenguajes internos para simplificar la compilación. Algunos de estos lenguajes son por ejemplo el lenguaje de árbol genérico o el SSA basado en Gimple.

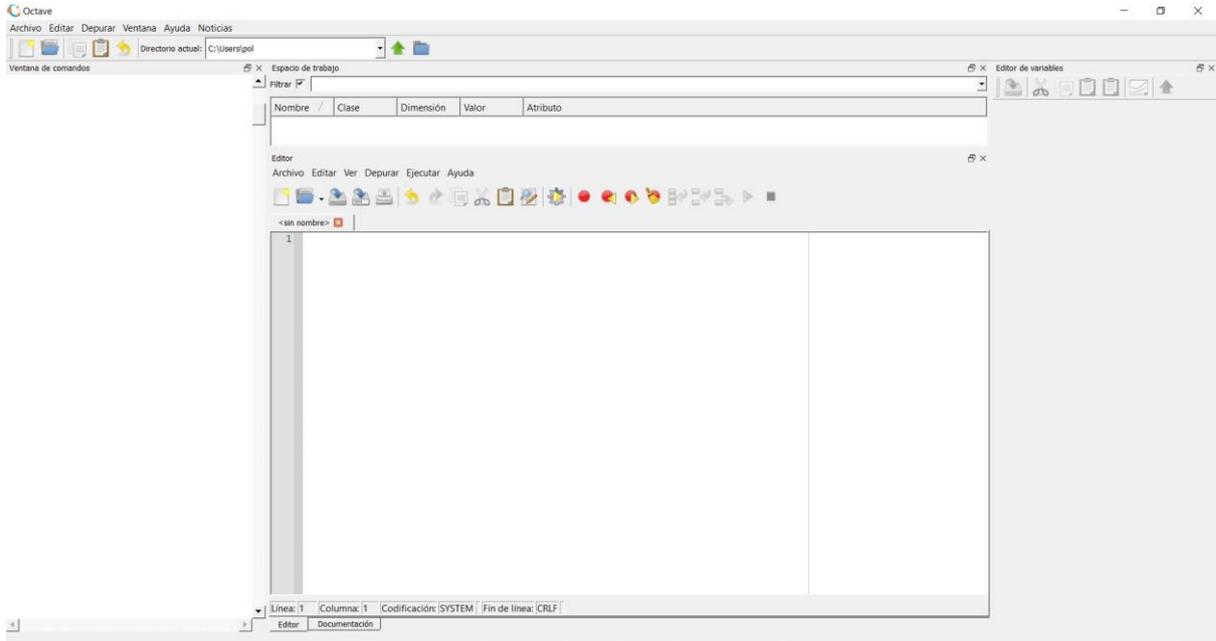
Finalmente ejecutará el código precompilado y se ejecutará por ejemplo el script.

El almacenamiento de este programa puede ser extendido utilizando módulos dinámicos.

A diferencia de otros programas, este tiene un intérprete que trabaja con otro basado en OpenGL con el cual puedes crear gráficos a partir de datos recopilados previamente.

## 1.7. Estructura

La estructura del programa es esta:



Podemos distinguir varias zonas diferenciadas:

- La barra de menús.
- La barra de herramientas.
- La ventana de comandos.
- El espacio de trabajo.
- El editor
- El editor de variables.

A continuación explicaré en detalle las diferentes opciones y características de cada parte.

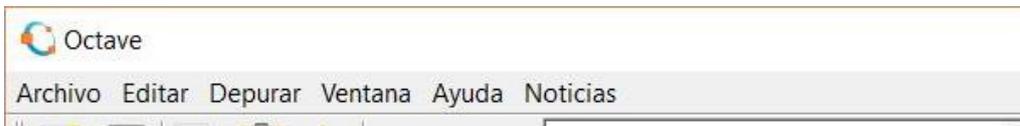
## 5.1.

### La barra de menús.

En la parte superior de la ventana principal tenemos la barra de menús con diferentes opciones.

Podremos encontrar en orden, como se ve en la imagen:

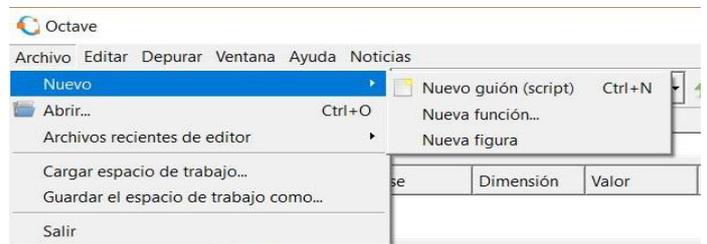
*Archivo, Editar, Depurar, Ventana, Ayuda, Noticias.*



En el apartado de Archivo hay diferentes subapartados como se puede ver en la imagen.

En el menú **Nuevo** podemos ver las diferentes opciones:

En primer lugar aparece la opción de **crear un nuevo script** (es el texto que escribimos para que cuando lo introduzcamos en el espacio donde se ejecuta, nos permita la ejecución del programa). Dentro de este mismo apartado también aparecen otros subapartados que no he usado porque yo me he centrado únicamente en los scripts.



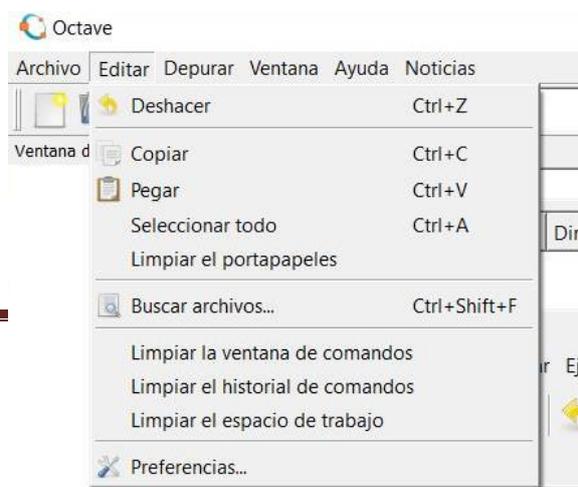
En segundo lugar aparece la opción de **abrir un script** almacenado en el ordenador.

En tercer lugar encontramos la opción de **abrir los archivos recientes** que has estado modificando en el editor.

En cuarto y en quinto lugar encontramos una opción que nos permite **cargar o guardar** nuestro espacio de trabajo eso quiere decir todos los scripts, funciones o figuras.

Y por último la opción de **salir** de octave.

En el menú **Edit** se pueden ver diferentes apartados:



Primero encontraremos la opción de **deshacer** cualquier error que hayamos cometido.

Después encontraremos la opción de copiar y seguidamente la de pegar.

Más abajo encontraremos la opción de seleccionar todo y si bajas un poco más aparecerá la opción de limpiar el portapapeles.

Si seguimos bajando encontraremos la opción de **Buscar archivos..**

También aparece la opción de limpiar la ventana de comandos, el historial de comandos y también el espacio de trabajo.

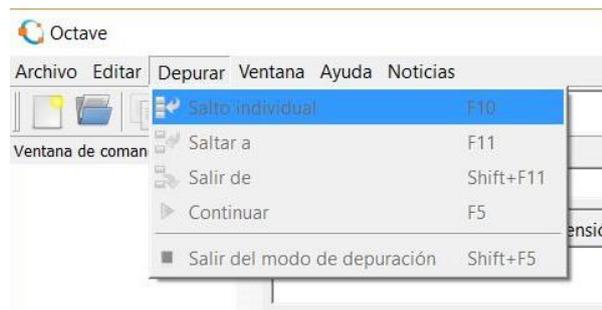
En este apartado también encontraremos la opción de modificar y configurar nuestras preferencias.

En el menú **Depurar** encontraremos varias opciones en la hora de ejecutar nuestro script por ejemplo:

*Salto individual*: nos salta una fila por orden.

*Saltar a* : saltara a la línea que tu le digas.

*Salir a* : saldrá de la fila que estamos trabajando.

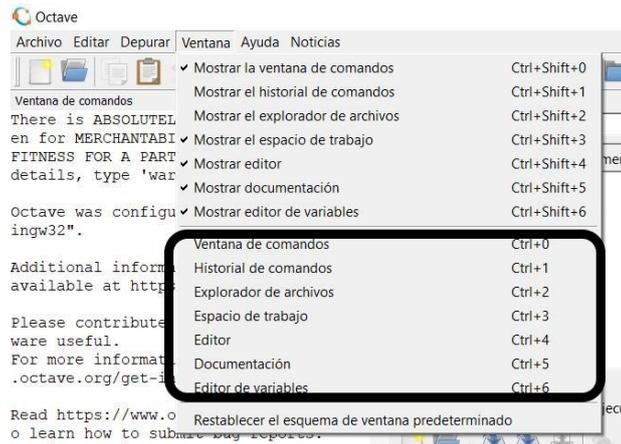


*Continuar*: nos permite seguir con la edición.

*Salir del modo de depuración* :nos permite salir de este modo.

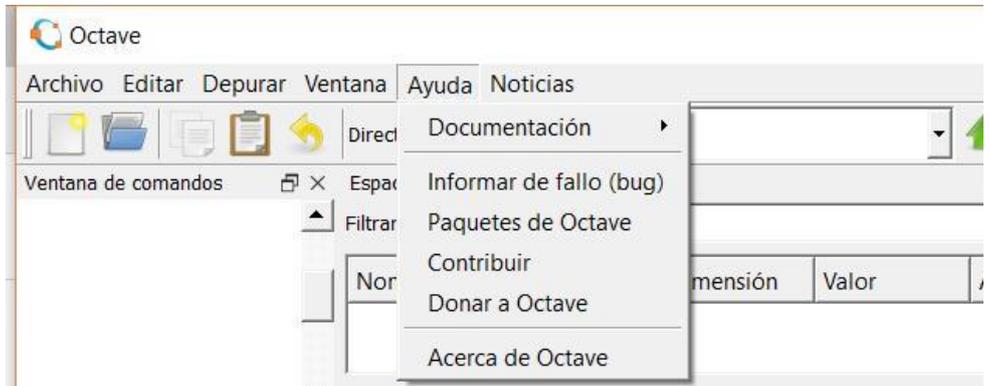
En el menú **Ventana** nos permitirá modificar lo que nos aparecerá en la pantalla. Desde este apartado podremos observar el historial o la ventana de comandos entre otros.

También puede mostrar cualquiera de las opciones de la segunda mitad (redondeadas).



En el menú de **Ayuda** tendremos varios apartados:

Primero tendremos el apartado de *documentación* desde donde podremos acceder a documentación en el caso de que tengamos alguna duda. También puedes acceder con la biblioteca de datos que viene ya instalada en nuestro programa o puedes acceder a foros de internet.



Después tendremos el apartado de

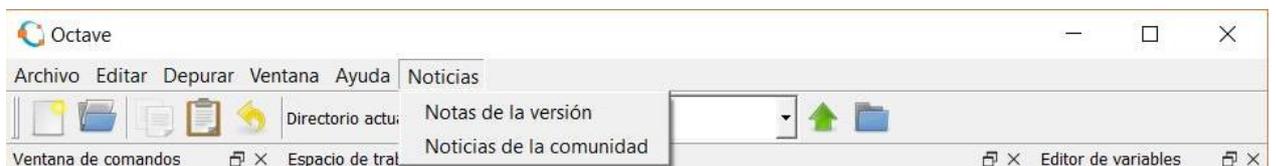
*Informar de fallo (bug)*: Aquí podremos informar sobre algún fallo del programa.

En el apartado de *Paquetes de Octave* podremos descargarnos las extensiones o *plugins*. del programa. Esta opción es muy útil para evitar que la carga del programa sea muy grande. Con esta opción puedes descargarte solo las extensiones que necesites desde la base de datos de Octave .

Las otras dos opciones las voy a obviar ya que las dos sirven para donar a Octave.

La última opción nos ayudará en caso de duda a entender más sobre Octave.

En el apartado de *Noticias* podremos acceder a las novedades y noticias relacionadas con Octave.



## La barra de herramientas.

Por debajo de los menús, encontramos unos iconos con las herramientas más habituales.



Estos iconos son los típicos que nos encontraremos en cualquier programa. En orden, serán: **Nuevo guion, Guardar, Copiar, Pegar y Deshacer.**

En esta parte podremos ver en qué directorio estamos trabajando. Por ejemplo yo estoy trabajando con mi usuario como se puede ver en la imagen pero puedes trabajar con cualquier directorio.

La flecha verde que apunta hacia arriba sirve para cambiar de directorio y la carpeta azul sirve para explorar tus directorios.



Por debajo de las barras de herramientas encontramos la zona de trabajo que tiene 3 partes principales: la *ventana de comandos*, el *espacio de trabajo* y el *editor de variables*.

## La ventana de comandos.

En la *ventana de comandos* podremos ejecutar nuestro script y ver como evoluciona. También podremos ejecutar algunos comandos que nos permitirán, por ejemplo, iniciar el programa o apagarlo. Y podremos hacer que aparezca el diccionario de palabras para crear nuestro script, por ejemplo.

```
Ventana de comandos
GNU Octave, version 4.4.0
Copyright (C) 2018 John W. Eaton and others.
This is free software; see the source code for copying conditions.
There is ABSOLUTELY NO WARRANTY; not even for MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. For details, type 'warranty'.

Octave was configured for "x86_64-w64-mingw32".

Additional information about Octave is available at https://www.octave.org.

Please contribute if you find this software useful.
For more information, visit https://www.octave.org/get-involved.html

Read https://www.octave.org/bugs.html to learn how to submit bug reports.
For information about changes from previous versions, type 'news'.

>> |
```

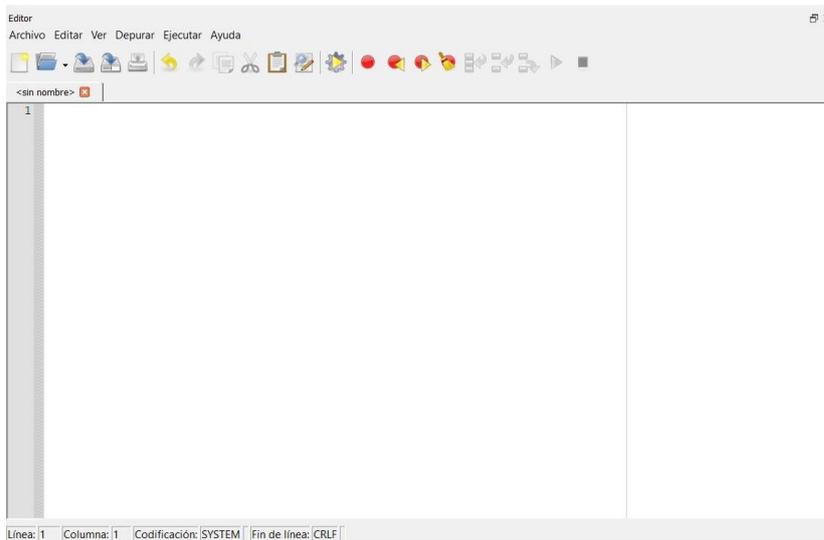
## El espacio de trabajo.

En el apartado de espacio de trabajo encontraremos todos los scripts que hemos realizado por orden. Se clasificarán en una tabla según: **nombre, clase, dimensión, valor y atributo.**

Esto se ve de esta manera:



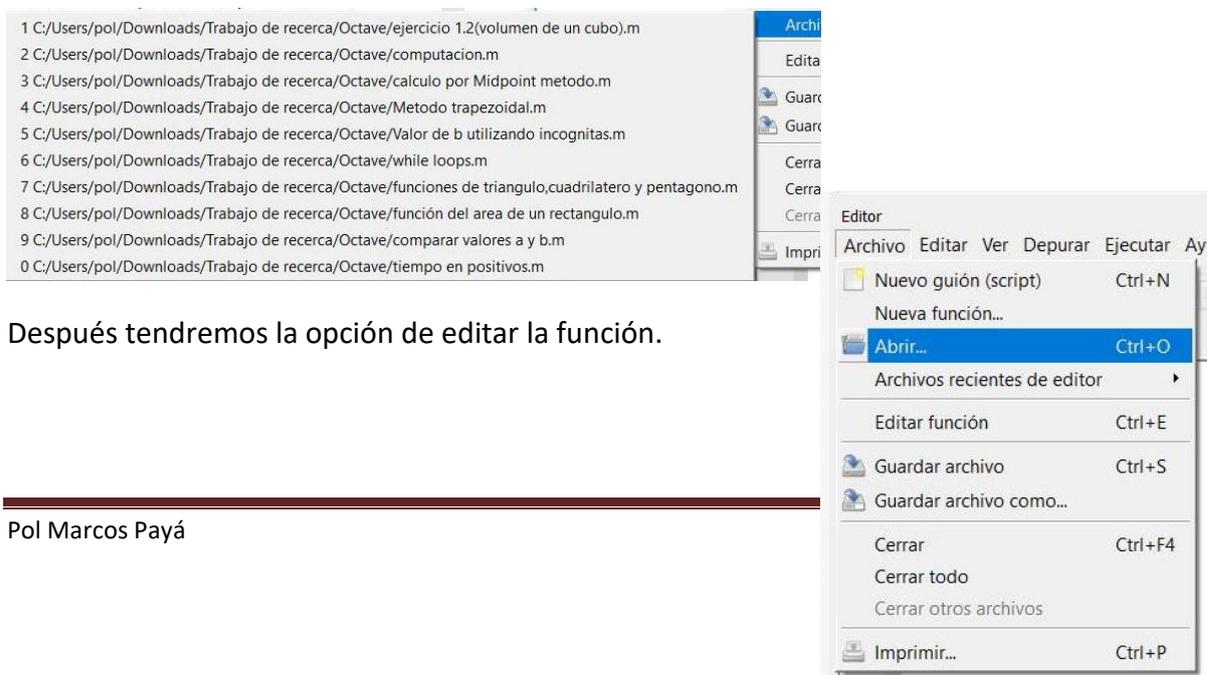
## El editor



En el *editor* encontramos muchas opciones. Las más destacadas son:

En la parte superior encontramos una serie de opciones similares a las de la parte de arriba aunque centradas en la edición.

Primero encontraremos la opción de *archivo*. Seguidamente la opción de crear un *nuevo script*, después encontraremos la opción de *abrir* un script guardado en tu ordenador y un poco más abajo encontraremos la opción de ver los *archivos recientes del editor*. Estos son mis últimos ejercicios que se guardaran allí en caso de que quiera volver a editarlos.



Después tendremos la opción de editar la función.

Más abajo encontraremos las opciones de *guardar y guardar como*.

Encontraremos la opción de cerrar el editor y después de cerrarlo todo.

Por último encontraremos la opción de *imprimir* nuestro script

En el apartado *editar* encontraremos:

La opción de *deshacer*. Esta nos será muy útil en el momento de editar ya que nos permite retroceder y cuando, por ejemplo, al editar un script te equivocas y borras una parte que necesitabas utilizando esta opción podrás volver a tenerlo.

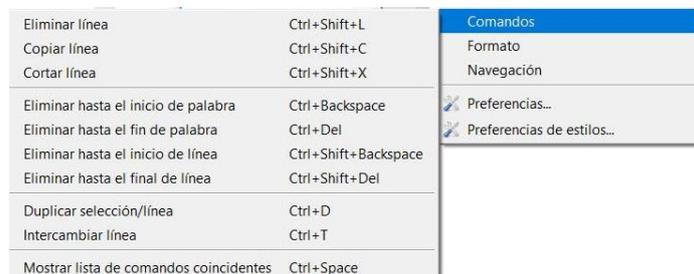
La siguiente opción es *rehacer*. Con esta opción lo que conseguiremos es volver a escribir un script que hayamos borrado.

En el siguiente grupo encontraremos las tres opciones típicas : *cortar ,pegar y copiar*.

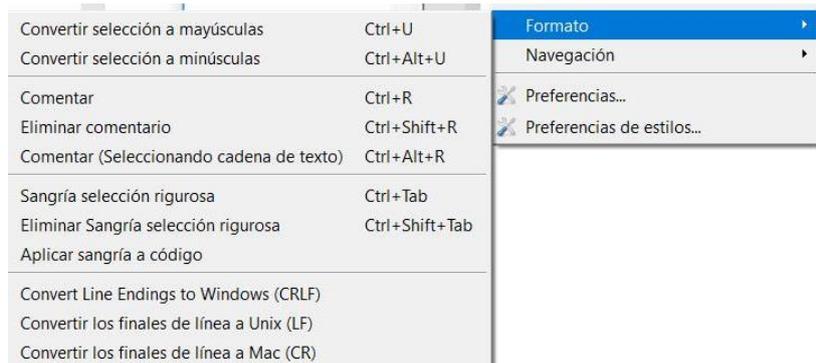
La opción de *seleccionar todo* nos permitirá seleccionar todo el script rápidamente.

Las siguientes cuatro opciones nos permitirán buscar archivos y como se puede ver en la siguiente opción también reemplazarlos.

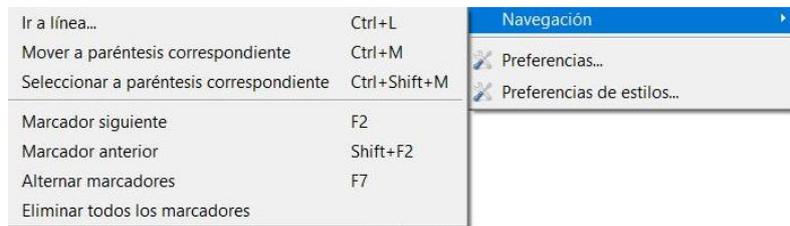
La opción de *comandos* nos permitirá varias cosas como se puede ver a continuación:



La opción de *formato* nos permitirá también un seguido de cosas como se pueden ver a continuación:

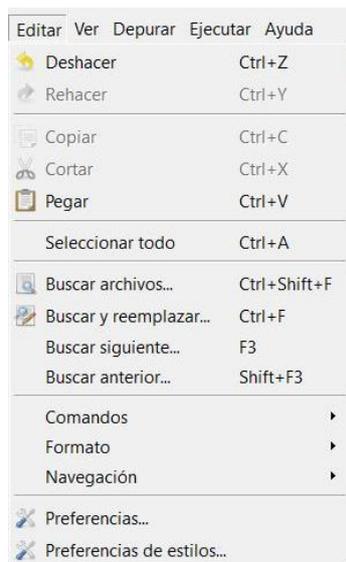


En la siguiente opción de *navegación* también encontraremos una serie de opciones como son:

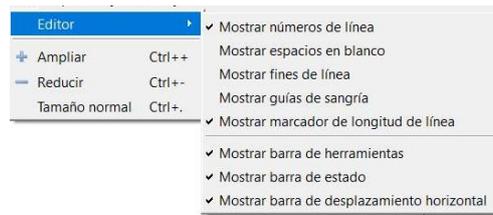


La siguientes dos opciones nos permitirán modificar las *preferencias* del editor y de los estilos de escritura.

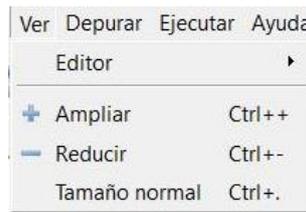
Así se ve todo lo que he explicado de forma general:



En el apartado de *ver* encontraremos primero la opción del *editor* donde se nos mostraran estas opciones:



En este apartado encontraremos también la opción de *reducir*, *ampliar* y verlo a *tamaño normal*.



En el apartado de *depurar* encontraremos las opciones de primero *alternar los puntos de interrupción* que nos permitirá agregar puntos de interrupción aleatoriamente.

La siguiente opción permitirá añadir un *punto de interrupción siguiente*, es decir, añadirá un punto a la siguiente fila que estés trabajando.

La opción de un *punto de interrupción anterior* añadirá un punto de interrupción a la fila que tengas detrás.

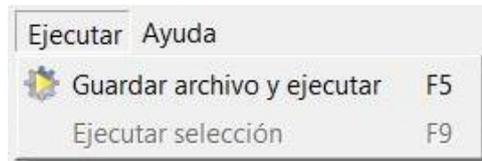
La opción que se muestra coloreada nos permitirá *eliminar todos los puntos de interrupción*.

Las siguientes opciones nos permitirán saltar de fila en fila de manera específica o en orden. Otra función será la de salir de la edición o continuar editando.

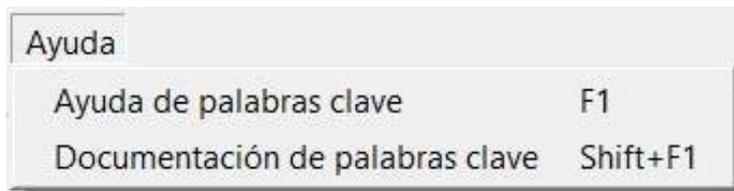
Por último encontraremos la opción de *salir del modo de depuración*.



El apartado de *ejecutar* encontraremos la opción de *guardar el archivo y ejecutar* y si seleccionamos una parte del script que queremos ejecutar existirá la opción de *ejecutar la selección*.



Finalmente encontraremos la opción de *ayuda* que nos permitirá consultar en todo momento las diferentes dudas o problemas que el programa nos plantee.



Debajo de este menú encontraremos esta serie de iconos.

Encontraremos por este orden:

**Nuevo guión, Abrir archivo, Guardar, Guardar cómo, Imprimir, Deshacer, Rehacer, Copiar, Pegar, Revisar, Guardar y ejecutar, Alterar los puntos de interrupción.**



Los siguientes cuatro iconos nos permiten saltar de manera individual o por orden.

Los otros dos nos permitirán salir de la depuración o continuar.



Los siguientes dos iconos nos permiten ejecutar y empezar de nuevo.



La última parte del *editor* y la más importante es donde escribimos nuestro script.



Abajo se puede ver como marca la línea y la columna donde estamos ubicados. En este caso como está en blanco nos la marcará por defecto (1,1).

También podremos ver la codificación que en este caso es la del sistema.

Y el fin de la línea que nos marcará CRLF.

### **El editor de variables.**

Por último encontraremos el *editor de variables* donde podremos *copiar, pegar y guardar* de manera general o *guardar como* como se muestra en la imagen.

También podremos editar la representación gráfica y cambiar de directorio con los otros dos emoticonos.



# PARTE PRÁCTICA

---

En esta parte práctica he combinado el trabajo personal en casa y con el ordenador con una serie de visitas y/o conferencias/entrevistas, unas presenciales como en el caso de la visita al superordenador y otras on-line como la entrevista con el Sr Vázquez-Poletti.

La parte práctica del trabajo se centrará en 4 puntos básicos que nos servirán para entender todo el proceso.

Empezaré con el análisis detallado de una serie de scripts explicando cada uno de los pasos dados y lo acompañaré de unos ejemplos que servirán de guía y soporte visual.

Continuaré explicando la visita a Barcelona Supercomputing Center y de los usos del superordenador dentro del campo de la investigación.

También detallaré la información extraída de la entrevista al Sr Vázquez-Poletti, ingeniero electrónico y doctor en la Universidad Complutense de Madrid.

Por último, en el apartado de los anexos, explicaré cómo cree un servidor de Wordpress y su instalación.

Terminaré con la práctica en la UB.

## 5. Programación de Scripts en Octave

A continuación voy a presentaros algunos de mis scripts realizados en Octave como parte práctica.

Para realizar estos scripts, usé un libro llamado “*Programming for computations*” que además de enseñarte a usar el programa te propone ejercicios para comprobar que has entendido cada apartado.

En este documento no he analizado todos los ejercicios que he realizado. He decidido escoger unos pocos que reflejan la evolución en el uso del programa y con eso su sintaxis. Empecé a hacer la parte práctica sin saber cómo utilizar el programa y sin un conocimiento previo sobre C++ y he conseguido aprender un poco sobre Octave.

Los ejercicios están clasificados por orden de dificultad para que podáis observar mi evolución en Octave y van desde lo más básico, como es calcular el volumen de un cubo, hasta la implementación de dos métodos de computación usados para computar integrales en un solo script.

### 1ª práctica: Volumen de un cubo

Es un script muy sencillo. Consiste en:

- a) asignarle un valor a una variable C.
- b) Plantear la ecuación para calcular el volumen de un cubo.



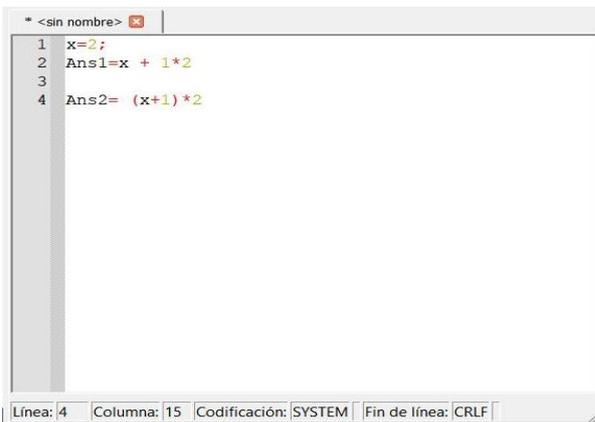
```
ejercicio 1.2(volumen de un cubo).m
1  % volumen de un cubo
2  C= 4
3  V= C^3
4
```

Línea: 4 | Columna: 1 | Codificación: SYSTEM | Fin de línea: CRLF

## 2ª práctica: Dos ecuaciones

En este script la dificultad es un poco más elevada ya que contaba con un poco más de “experiencia”

- a) Se asigna el valor 2 a la variable.
- b) Se plantean dos ecuaciones para obtener dos resultados diferentes.



```

1 x=2;
2 Ans1=x + 1*2
3
4 Ans2= (x+1)*2
    
```

Línea: 4    Columna: 15    Codificación: SYSTEM    Fin de línea: CRLF

Como los siguientes 3 ejercicios son muy parecidos pasaré directamente a otro apartado donde compararé valores para obtener resultados distintos.

## Práctica 3: Comparar valores a y b

a) Primero lo que hacemos es asignar unos valores a las variables **a** y **b**, (asigno a un valor más alto que **b**).

b) Asignaremos una condición como podemos ver en la tercera, quinta y séptima fila

En la tercera le estamos diciendo que si **a** es más grande que **b** debe escribir como solución que ‘**a** es el más pequeño de los dos’, en cambio si las dos variables son iguales debe escribir que ‘**a** y **b** son iguales’.

c) Si la variable **b** es más grande que la **a** debe escribir que ‘**a** es el número más grande de los dos’.

```

* <sin nombre> x
1 a = input('dale un numero a: 6');
2 b = input('dale un numero b: 5');
3 if 6 < 5
4 fprintf('a es el más pequeño de los dos\n');
5 else if 6 == 5
6 fprintf('a y b son iguales\n');
7 if 6 > 5
8 fprintf('a es el numero más grande de los dos\n');
9 end
10
11
12
13 %Este programa ayuda a escoger cual de los dos valores es más grande igual o más pequeño
    
```

Línea: 13 | Columna: 89 | Codificación: SYSTEM | Fin de línea: CRLF

#### Práctica 4: Función como argumento

```

* <sin nombre> x
1 function function_as_argument (x,y)
2
3     x=10
4     y=12
5
6     area1_triangulo_xy_handle = @area1_triangulo_xy;
7     area2_cuadrilatero_xy_handle = @area2_cuadrilatero_xy;
8     area3_pentagono_xy_handle = @area3_pentagono_xy ;
9
10    area1 = treat_xy (area_triangulo_handle , x , y);
11    fprintf('%f\n',area1);
12    area2 = treat_xy (area_cuadrilatero_handle , x , y);
13    fprintf('f\n',area2);
14    area3 = treat_xy(area_pentagono_handle,x , y);
15    fprintf ('f\n',area3);
16 end
17 function result =area1_triangulo_xy (x,y);
18     area_triangulo= (y*x)/2;
19 end
20 function result =area2_cuadrilatero_xy (x,y);
21     area_cuadrilatero = x*y;
22 end
23 function result =area3_pentagono_xy (x);
24     area_pentagono= ((5*x)*4.89)/2;
25 end
26
    
```

Línea: 26 | Columna: 1 | Codificación: SYSTEM | Fin de línea: CRLF

a) En esta primera parte planteamos una función con las variables **x** e **y**.

(Más adelante les asignaré un valor determinado para poder realizar las ecuaciones).

```

function function_as_argument (x,y)

    x=10
    y=12
    
```

b) A continuación asignaré un nombre y un número a cada área de las 3 figuras:

-el nombre de **area1**, **area2** y **area3**. A continuación, y mediante **fprintf**, le asignaremos números reales con punto fijo con el símbolo (%f) y con el otro símbolo lo que hacemos es asignar números reales (n') y lo repetiremos con cada una de las áreas como se puede observar.

```
function function_as_argument (x,y)

    x=10
    y=12

    area1_triangulo_xy_handle = @area1_triangulo_xy;
    area2_cuadrilatero_xy_handle = @area2_cuadrilatero_xy;
    area3_pentagono_xy_handle = @area3_pentagono_xy ;
```

Con esto lo que haremos es realizar una función y lo indicaremos con la palabra *función* arriba y para marcar el final utilizaremos la palabra *end*.

```
function function_as_argument (x,y)

    x=10
    y=12

    area1_triangulo_xy_handle = @area1_triangulo_xy;
    area2_cuadrilatero_xy_handle = @area2_cuadrilatero_xy;
    area3_pentagono_xy_handle = @area3_pentagono_xy ;

    area1 = treat_xy (area_triangulo_handle , x ,y);
    fprintf('%f ,n',area1);
    area2 = treat_xy (area_cuadrilatero_handle , x ,y);
    fprintf('f ,n',area2);
    area3 = treat_xy (area_pentagono_handle , x ,y);
    fprintf ('f,n',area3);
end
```

d) Finalmente lo que hacemos es plantear tres funciones con el mismo método:

marcar el inicio de función con la palabra *function* y con la palabra *end* finalizarlo, en cada una de las áreas que queremos calcular.

```
function result =area1_triangulo_xy (x,y);
    area_triangulo= (y*x)/2;
end
function result =area2_cuadrilatero_xy (x,y);
    area_cuadrilatero = x*y;
end
function result =area3_pentagono_xy (x);
    area_pentagono= ((5*x)*4.89)/2;
end
```

### Práctica 5: While loop

En este script lo que hacemos es plantear un bucle, esto significa que una parte del programa es ejecutada un número de veces en sucesión.

Los pasos a seguir son:

a) Asignar un número, en mi caso escogeré el número 10, después asignaré el valor de  $n$  a  $i$ .

b) Establecer un valor  $n$  que tiene que ser mayor que 1 ( $n > 1$ ). Lo que conseguimos haciendo servir la función While es asignar que, cuando  $n$  valga más que 1, ejecutamos las ecuaciones de abajo que nos indican que  $n = n - 1$  (le restamos una unidad a  $n$ );  $i = i * n$  (multiplicamos el valor imaginario por  $n - 1$ ) e indicaremos el final con la función `end`.

c) Al Ejecutarlas generamos un bucle que nos genera un número. En este caso  $n = 3628800$ .

```
* <sin nombre> [x]
1  n=10;
2  i =n;
3  while n >1
4      n=n-1;
5      i= i*n;
6  end
7
8  disp(['n! = ' num2str(i) ])
9
```

### Práctica 6: Valor de b utilizando incógnitas

En este script lo que hacemos es mediante una función en bucle asignarle un valor a  $b$ , a partir de los siguientes pasos:

a) Asignar unos valores a  $r$  y  $a$ , e indicar que  $i$  es igual al valor que queremos conseguir.

b) Plantear la ecuación. En nuestro caso, es la ecuación para calcular el área de un círculo ( $A = \pi * r^2$ ).

c) Indicar que  $i = 1$  y seguidamente plantear la función del bucle utilizando la función *While*.

d) Escribir que  $b$  ha de ser más pequeño que la ecuación de la circunferencia ( $i(b) < \pi \cdot r^2$ ) y en la otra ecuación se le suma un uno a  $i$  ( $i = i+1$ ), finalmente indicar el final del bucle con la función *end*.

(Indicaremos que nos enseñe el valor que resulta (`fprintf ('the value of b(from positive): %f\n', b(i))`) y *fprintf* nos mostrará el resultado de la función,  $n'$  significa que nos mostrará valores de números naturales).

```
* <sin nombre>
1 r=10.6;
2 a=1.3;
3 i=b;
4
5
6 A=pi*r^2;
7
8 i=1;
9 while i(b) < pi*r^2
10     i = i+1;
11 end
12
13 fprintf ('the value of b(from positive): %f\n',b(i));
14
15
16
```

### Práctica 7: Método trapezoidal y Midpoint

He juntado estos dos métodos en un solo apartado ya que son dos de los métodos principales para computar integrales en un solo script.

a) Plantear las variables. Primero planteamos la función de arriba ( $\exp(-y^2)$ ) y a continuación, daremos valor a las variables  $a$  y  $b$ .

b) Utilizando la función (`fprintf('n midpoint trapezoidal\n');`) indicaremos lo que vamos a hacer. Esta función es equivalente a “printf” y sirve para que aparezcan argumentos opcionales a los que saldrían normalmente si no utilizáramos esto.

c) Utilizaremos *for* para crear un bucle con las condiciones que se indican a continuación:

-for  $i = 1:20$  (le asignaremos al valor  $i$  que sea igual a  $1/20$  y no podrá pasar de esto)

- $n = 2^i$ ; (para  $n$  haremos que se multiplique por dos el valor de la variable  $i$ )

- $m = \text{midPoint}(g, a, b, n)$ ; (asignamos las condiciones de nuestro método midpoint poniendo como puede verse las variables  $g, a, b, n$ )

- $t = \text{trapezoidal}(g, a, b, n)$ ; (asignamos las condiciones como en el método midpoint)

-`fprintf('%7d %.16f %.16f\n', n, m, t)`; (con esto lo que haremos es crear los parámetros para la respuesta y lo que queremos que contenga nuestra respuesta)

Con `end` ponemos fin a nuestro bucle.

```
1 g = @(y) exp(-y*2);
2
3 a = 0;
4
5 b = 2;
6
7 fprintf('    n    midPoint    trapezoidal\n');
8
9 for i = 1:20
10
11     n = 2^i;
12
13     m = midPoint(g, a, b, n);
14
15     t = trapezoidal(g, a, b, n);
16
17     fprintf('%7d %.16f %.16f\n', n, m, t);
18
19 end
```

## 6. Barcelona Supercomputing Center

Es un centro de investigación donde se forman a más de trescientos científicos (alumnos de la propia universidad). También es un centro de servicios de supercomputación dedicado a toda la comunidad científica que gestiona la Red Española de Supercomputación.

Es conocido por tener el superordenador *Mare Nostrum*, uno de los superordenadores del centro nacional de Supercomputación y el más potente de España.

Actualmente es el tercero más potente de Europa, aunque el decimosexto del mundo.

En noviembre de 2004 se clasificó como cuarto del mundo (se hacen dos clasificaciones al año en todo el mundo) y ha sido al puesto más alto que ha llegado desde su construcción (los primeros puestos son ocupados por países como China o Estados Unidos).

Este superordenador se ha ido actualizando llegando a tener 4 nombres diferentes:

El primer nombre que se le dio fue *MareNostrum1* en marzo de 2004 y esta versión fue creada con componentes de IBM.

Su capacidad de computación era de 42,35 Teraflops. Este volumen puede traducirse a 42,5 trillones de operaciones por minuto, una capacidad muy superior a cualquier ordenador de sobremesa que podemos poseer.



La segunda versión fue actualizada en noviembre del 2006 y pasó a llamarse *MareNostrum 2*. Básicamente se aumentó la capacidad de computación doblando la anterior y consiguiendo una capacidad de computación de 94,31 Teraflops. Esto se consiguió aumentando a 10240 el número de procesadores (antes tenía 4812).



La tercera versión fue el *MareNostrum 3* actualizada en 2012-2013. En esta actualización no se añadieron más nodos sino que se instalaron 48896 procesadores Intel Sandy Bridge conectados con 3056 nodos (ordenadores que no trabajan de manera individual sino que trabajan creando uno) en un bloque. En otro bloque se introdujeron 84 Xeon Phi 5110p conectados en 42 nodos y un disco sólido GPFS de más de 115 Tb y 2 Pb de capacidad .

Con estas actualizaciones se consiguieron 1,1 Petaflops de capacidad de computación.



Imagen 5: Actual Superordenador

Por último tenemos la versión actual *MareNostrum 4* que se divide en dos bloques diferentes:

1. Tenemos un primer bloque al que llamaremos *bloque de propósito general* que contiene 48 estanterías con 3456 nodos en total. Cada nodo contiene dos **Intel Xeon Platinum** chips cada uno con 24 procesadores (si hacemos el cálculo tenemos un total de 16588 procesadores con una memoria de 390 Terabytes).

Produce una potencia computacional de 11,15 Petaflops (más de 11.000.000.000.000.000 por segundo) 10 veces más que el *MareNostrum 3* y cabe destacar que no consume mucho más que su anterior máquina, solo hace que consuma un 30% por ciento de energía (1.3 MWatt/year).

2. El segundo bloque está formado por una conjunción de tres tecnologías que se han usado en otros superordenadores. Estas tecnologías son actualizadas constantemente por las grandes superpotencias a nivel computacional como son China o Estados Unidos.

-un procesador IBM power 9 unido con una NVIDIA GPUs, (los mismos componentes que contienen los superordenadores Sierra en Estados Unidos). Concretamente un proyecto que se está llevando a cabo con esta potencia es el proyecto 'Mino tauro'.

- unos procesadores Intel Knights Hill, también ubicados en Estados Unidos.

- unos procesadores de 64 bits ARMv8 que se encuentran en Japón.

Con esto lo que se pretende conseguir es probar cuál de estas tres formas de computación funciona mejor en estas instalaciones para instalarlas en un futuro.

El superordenador posee una capacidad de almacenamiento de 14 Petabytes y conectado con las infraestructuras del BSC tendrá en total una capacidad de 24,6 Petabytes. Estará conectada por unos cables de alta velocidad que conectarán todos los componentes de cada superordenador, en total conectará 3456 ordenadores y para ello se necesitará una gran cantidad de cable.

También estará conectado con otros centros europeos de investigación.

## 5.2. Software

Este ordenador trabaja con varios tipos de Software dependiendo de la parte del ordenador en la que te fijas. A continuación voy a mencionar y explicar un poco sobre cada tipo software:

El sistema operativo principal del sistema es el SLES 12 SP2. Este sistema operativo basado en Linux Kernel y desarrollado por SUSE fue creado en el 2000 y a diferencia de otras extensiones de Linux, esta se paga.

Esta extensión generalmente se usa para gestionar servidores.

Aunque nosotros lo utilizamos para poder hacer funcionar todas las máquinas que trabajan de manera independiente y simultáneamente consiguiendo unir toda capacidad de computación en una sola, lo cual nos proporciona un gran volumen de computación.

El sistema de cada una de las máquinas es el xCAT. Este sistema operativo está creado por Linux y lo que nos proporciona es una interfaz que engloba a todas las máquinas del superordenador y permite controlar su hardware.

Este software permite administrar de forma remota los sistemas, haciendo de esta una característica muy importante. También nos permite compilar todo el volumen de datos, es decir, traducir todas las operaciones realizadas en lenguaje binario y convertirlas en datos

Para ello utiliza dos tipos de compiladores:

-Los **Intel Cluster Studio** que tienen una gran capacidad para optimizar los códigos para las CPUs que se usan en el superordenador.

-Los **GNU compilers** los cuales se complementan perfectamente con los de Intel.

En el software del superordenador encontraremos también un proyecto abierto de **OpenMPI** (interfaz de paso de mensajes). Este proyecto es desarrollado por muchos de los superordenadores que pertenecen al top 500 mundial.

Toda esta tecnología busca una mejora en la implementación del MPI. También se quiere proporcionar un gran rendimiento y una mayor competitividad en todos los superordenadores y con esto involucrar a toda la comunidad de supercomputación, consiguiendo proporcionar una plataforma estable para la investigación.

Podremos encontrar a nivel de software una colaboración con Intel en la creación de una biblioteca para crear, mantener y probar aplicaciones avanzadas y complejas que solo se pueden probar en ordenadores con gran capacidad de computación.

Con esto se consigue que Intel mejore su software y también mejora el del MareNostrum. Y con la integración de la CPU se consigue mejorar el coste, la potencia y la densidad. También se consigue mejorar el ancho de banda del nodo y con ello mejorar la latencia de comunicación significativamente.

Gracias a la optimización del host se consigue ampliar el número de mensajes que se pueden enviar.

Y por último gracias a su estructura se consigue reducir el tiempo en el que se tarda en detectar un error y mejorarlo.



Imagen 6: Actual superordenador

### 5.3. Uso del superordenador

El superordenador no es solo usado por la universidad, de hecho solo lo usan un 4% del personal universitario, es decir, es usado por 226 investigadores.

Con un porcentaje de uso de un 16% encontramos a la Red Española de Computación que desarrolla más o menos 163 actividades anuales.

Con un 80% de uso está la PRACE (Partnership for Advanced Computing in Europe) que acapara prácticamente todas las horas de computación del superordenador.

También es utilizado por investigadores que tienen que presentar un proyecto y si ganan pueden usar el superordenador unas horas determinadas.

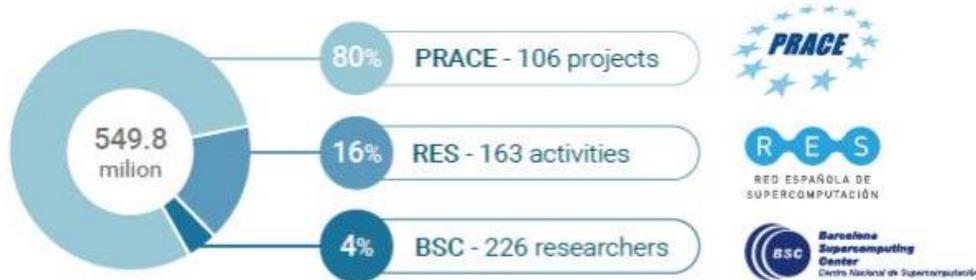


Imagen 7: Porcentaje de uso del BSC

### 5.4. Usos

Los proyectos que se realizan giran sobre:

-**Composición atmosférica.** Son proyectos que permiten investigar sobre la calidad del aire para desarrollar modelos de calidad del aire, como herramientas que permitan identificar las fuentes y procesos que determinen la calidad del aire para así poder predecir los episodios de contaminación. También hay proyectos que estudian la interacción de los aerosoles en el aire entre otras cosas.

- **Big Data.** En estos proyectos se estudia la interacción de las personas con las computadoras y a partir de los datos crear herramientas visuales y algorítmicas para analizar y estudiar grandes volúmenes de datos.

-**Bioinformática.** Es uno de los proyectos principales que intentan integrar, almacenar y transmitir un gran volumen de datos.

**-Biomecánica.** Se desarrollan básicamente simulaciones de órganos y también de inteligencia artificial.

**-Predicción climática** para la gestión de la agricultura y el agua, el pronóstico oceánico, el estudio de los ciclones tropicales o el estudio para saber dónde es más eficiente instalar un molino de viento (este último fue un proyecto encargado por Endesa).

**-Simulación social.** Aquí se estudian con un gran número de variables, simulaciones de la sociedad para conocer la evolución cultural, la seguridad etc.

## 7. Entrevista a José Luis Vázquez-Poletti

José Luis Vázquez-Poletti es profesor de la facultad de Informática de la Universidad Complutense de Madrid. Es el encargado del departamento de Arquitectura de Computadores y Automática. Pertenece al Grupo de Arquitectura de Sistemas distribuidos de esa misma universidad.

Recibió su máster en Ingeniería Informática en 2004 por la Universidad Pontificia de Comillas y posteriormente su posdoctorado en Arquitectura Informática en 2008 por la Universidad Complutense de Madrid donde realizó su proyecto de final de carrera colaborando con el CERN.

Actualmente está involucrado en varios proyectos europeos dedicados a la computación en la nube. Pero no solo ha trabajado con la computación en la nube, anterior a la utilización de esta, trabajó con un sistema de computación llamado Grip que trabaja de manera diferente a lo que a computación en la nube se refiere.

El proyecto en el que ha sido más conocido, basándonos en su número de citas es el proyecto de iCanCloud. Este proyecto se basa en el cálculo del número de máquinas necesarias (dependiendo del volumen de trabajo) y el límite de tiempo en el que se tiene que realizar el trabajo.

### 1. ¿Qué es la computación en la nube?

Es un modelo de provisión de servicios de computación. Un modelo de provisión es lo que nosotros en casa denominaremos electricidad o gas... Pero nosotros lo relacionamos con la computación.

De la nube por ejemplo se puede obtener almacenamiento (como se utiliza en dropbox o drive).

De otro modo lo que se consigue con la computación en la nube es disponer de un conjunto de máquinas con una potencia realmente superior a lo que puedes estar utilizando día a día para satisfacer picos de trabajo cuando tú lo necesites o desees.

Si se tuviera que describir con 3 características la computación diríamos que es:

- Bajo demanda, es decir que dispones de ella cuando quieras.

-Dinámica, es decir que sepa reaccionar en los momentos que tú necesitas algo y te lo pueda suministrar.

-Elástica, es decir, que tú infraestructura pueda variar dependiendo del flujo de trabajo contratando un volumen más grande o más pequeño de máquinas.

## **2. ¿Qué ventajas le proporciona trabajar con los servidores de Amazon?¿Trabaja con otro proveedor?**

Para empezar Amazon es un proveedor de servicios en la nube que se denomina servidor público, es decir, tú pagas por el tiempo de uso de sus servicios, en este caso el alquiler de sus máquinas.

Lo he elegido ya que nos proporciona una mejor relación de calidad-precio y nos proporciona una gran cantidad de máquinas diferentes dependiendo de la demanda y también buenos medios de almacenamiento a un muy bajo precio relativamente.

No, ahora trabajamos solo con Amazon por estas razones y porque de momento nos ofrecen las mejores condiciones y resultados en el área de aplicaciones.

## **3. ¿Cómo empezó a utilizar la computación en la nube?¿Por qué ?**

Empecé con mi tesis doctoral y utilicé un modelo previo a la computación en la nube, la computación Grid (basada en la unión de muchos ordenadores alrededor del mundo para crear un superordenador. Cada uno se reparte un seguido de tareas y así lo consiguen).

Y pude realizar una colaboración con el CERN. Ellos utilizaban este tipo de computación para procesar los datos y poder almacenarlos ya que necesitaban un gran volumen de computación y de almacenamiento para su nuevo acelerador de partículas (LHC).

Y luego empezamos a utilizar lo que era en ese momento la tecnología del futuro para investigación y gracias a la computación pudimos ahorrarnos mucho tiempo y recursos ya que como ya he comentado antes, puedes disponer de estos servicios cuando quieras y con el volumen de máquinas que quieras, sin tener la necesidad de comprarlas.

Lo utilizamos ya de manera más profunda para la creación de una aplicación asociada con la primera misión de Marte. Era una colaboración con Rusia y Finlandia y

empezamos a utilizar Amazon por su capacidad de computación y sobre todo por que debíamos hacerlo en un tiempo límite.

#### **4. ¿Por qué eligió trabajar con Octave en MODELING AND SIMULATION OF THE ATMOSPHERIC DUST DYNAMIC: FRACTIONAL CALCULUS AND CLOUD COMPUTING?**

Decidimos utilizar Octave porque teníamos que analizar los datos que nos llegaban de la sonda y analizamos, por ejemplo, que había una gran cantidad de radiación que sería nociva para nosotros.

Actualmente con Octave y los datos que nos llegan lo que estamos intentando es, descubrir alguna zona que nos salvaguarde de manera relativa de la radiación.

Principalmente lo usamos ya que es un software gratuito y trabajar con Matlab hubiera sido un problema de licencias ya que necesitaríamos muchas licencias y muchas máquinas

#### **5. De todos los proyectos en qué ha trabajado, ¿cuál ha sido el que le ha exigido más en cuanto a computación?**

El que más me ha exigido cuando yo me empecé a adentrar en el mundo de la computación fue una aplicación de física de fusión en qué se necesitaban realizar unos cálculos bastante grandes para predecir la trayectoria de las partículas del disparador de microondas (usado para encender el plasma del reactor). El objetivo era descubrir cuál era la mejor configuración para que produjera más energía pero con menor consumo.

Y la que más potencia de cálculo es una colaboración con la Agencia Estatal de Meteorología donde queríamos ver si era mejor utilizar la computación en la nube y dejar sus servidores libres para investigación. Para ello lo que hicimos fue dividir la atmósfera en pequeños cubos de varios metros y por cada cubo calcular sus interacciones y mi trabajo fue calcular el volumen de computación que necesitaban.

#### **6. ¿Qué consejo me podría dar para adentrarme en el mundo de la computación basándonos en su experiencia?**

Yo te diría que nunca dejes de aprender y siempre tengas curiosidad en lo que haces y tengas claro lo que quieres llegar a ser.

**7 .De todos los motivos por los que decide utilizar la computación en la nube en lugar de superordenadores ¿cuál es el que le parece más importante?**

El económico es importante pero el que realmente es más útil es poder disponer bajo demanda de unos recursos de forma dinámica, es decir, cuando los necesites.

**8. ¿Cómo cree que la computación en la nube puede ayudar al ser humano ahora? ¿Y en un futuro?**

Puede ayudar al ser humano principalmente en el tema espacial, en el uso de las naves espaciales haciendo que muchas de las funciones queden automatizadas y así ayudando a que no se cometan errores.

**9.¿Utilizar la computación en la nube es seguro en el sentido de que es más difícil de hackear ?¿O está igual de “ desprotegido” trabajando en un ordenador?**

Está igual de desprotegido, lo único que te puede salvar es que estas dentro de una multitud de máquinas virtuales las cuales están alojadas en recursos físicos.

**10.¿Qué proyecto de final de carrera me recomendaría realizar si quiero enfocar mi vida profesional en el ámbito de Defensa?**

Es ver el entorno que te rodea y ver los diferentes escenarios en los cuales podrás desarrollar tu proyecto, hablar con gente que ya ha pasado por eso.

Y ver quien te supervisará y te ayudará.

## 8. Conclusiones

Una vez terminado el trabajo puedo decir que he cumplido la mayoría de los objetivos que me había planteado al principio:

-Aprender a usar un detector TimePix para obtener medidas de diferentes tipos de radiación en diferentes experimentos. En un principio este iba a ser uno de mis objetivos principales pero luego decidí centrarme en la computación.

-Usar el software PixetPRO para configurar el detector y obtener los datos fue uno de los primeros objetivos a cumplir, con la práctica que realizamos en el laboratorio de física.

-Aprender cómo las partículas y los rayos X viajan a través de materiales mediante varios experimentos. Este apartado ha sido trabajado por mis compañeros y por mi en la práctica que hicimos en la UB. Sin embargo, no he profundizado en este aspecto tanto como algunos de mis compañeros por haber orientado mi trabajo en otros campos.

-Aprender a utilizar herramientas para crear una página web y aprender a gestionarla, instalar y configurar un servidor local de APACHE con PHP y una base de datos MySQL e instalar un sistema de gestión de contenidos (WORDPRESS) sobre el servidor local son 3 objetivos que he desarrollado al crear una página web para comunicar los resultados de las actividades realizadas con el detector y administrarla de manera adecuada. Aunque esta página web no tiene la complejidad y profundidad que me hubiera gustado darle, cumple su función y sobre todo me ha permitido aprender las nociones básicas de creación y administración de la página, por lo que en un futuro podré desarrollar páginas más complejas y con más funcionalidades.

-Aprender cual es la estructura y el funcionamiento de un programa de análisis numérico (OCTAVE) fue uno de los objetivos que desarrollé en verano, siguiendo las directrices del libro *Programming for Computations*. Este objetivo fue completado al mismo tiempo que me dedicaba a aprender la sintaxis del lenguaje de programación de Octave para realizar pequeños scripts de algoritmos de análisis numéricos. Dada la complejidad de Octave y mis pobres conocimientos de las matemáticas aplicadas a la computación, este objetivo no se ha cumplido tal y como esperaba. De hecho, aunque he podido aprender la sintaxis de OCTAVE a un nivel bastante satisfactorio, la dificultad de las matemáticas involucradas en la mayoría de los algoritmos de análisis numérico aplicado a la ciencia ha sido el factor más limitante para poder completar este objetivo.

-Esta dificultad no me impidió saber que es el análisis numérico, cuáles son sus partes y características (ecuaciones, métodos....) y su uso en ciencia e investigación, pero sin

embargo no pude profundizar demasiado en este objetivo por la dificultad matemática que tiene.

- Sin embargo si que he podido descubrir las limitaciones impuestas por el error en los tratamientos numéricos aproximados y como tratar con dichos errores, aunque haya sido de manera superficial y también he podido investigar que es la estabilidad numérica para así poder comprender su importancia en el análisis numérico.

-Otro de los objetivos que no pude cumplir fue profundizar en el funcionamiento de PixetPRO haciendo una entrevista con algunas de las personas que se dedican a programarlo ya que no fue posible contactar con nadie que cumpliera con este perfil.

-Uno de los objetivos que más me ha sorprendido ha sido investigar qué soluciones se han encontrado para analizar datos en proyectos científicos con gran análisis de datos. La variedad de las soluciones y los proyectos de ciencia puntera a los que se aplica (genoma humano, síntesis de drogas, física de cuántica, exploración espacial, etc.) me han resultado muy interesantes.

-La posibilidad de entrevistar a José-Luis Vázquez Poletti, ingeniero informático de la UCM, me ha permitido comprender la importancia de la tecnología de la información y el análisis de los datos en el desarrollo científico actual, al tiempo que me ha permitido conocer a un científico de primer orden. Es realmente motivador poder conversar con personas punteras en su campo que no dudan en dedicar algo de su tiempo a divulgar sus conocimientos entre los estudiantes como yo.

-Otra parte bastante sorprendente fue investigar la organización y el sistema de trabajo de proyectos científicos que utilicen supercomputadores para realizar investigaciones. La visita al Barcelona SuperComputing Center fue muy interesante. Poder ver en vivo el Mare Nostrum, uno de los ordenadores más potentes del mundo y aprender de primera mano sus características, funcionamiento y conocer los proyectos en los que se utiliza me permitió comprender mejor como es el trabajo científico y como la computación contribuye a su desarrollo.

-Aunque no era uno de mis objetivos de partida, conocer qué tipos de almacenamiento y organizaciones de trabajo se usan en computación aplicada a análisis de datos científicos (cloud computing,grid.....) fue un aspecto que apareció al comenzar mi investigación y, sobre todo, tras la entrevista con el doctor Vazquez-Poletti, por lo que lo incorporé a mi trabajo.

Aparte de estos objetivos he aprendido también a hacer una videoconferencia por Skype, he conocido a personas que están desempeñando una labor muy importante en el ámbito científico-tecnológico y me he dado cuenta del poco valor que se le da a la tecnología y a la cultura general (ayudas, subvenciones...) en un mundo totalmente tecnológico. Pero este es ya otro tema...

También me ha servido para tener un poco más claros mis objetivos universitarios a corto plazo y a poder decidirme, con más información, sobre qué carrera quiero estudiar.

En definitiva, este trabajo ha superado las expectativas que me había marcado en un principio.

Agradezco la ayuda desinteresada de mi madre, Esther Payá por ayudarme a corregir y mejorar el lenguaje de mi trabajo y de mi tutor de TR, Daniel Parcerisas por ayudarme en todo.

También quiero agradecer la ayuda de profesionales del sector como Rafa Ballabriga por dejarnos el detector y facilitarme los contactos, José-Luis Vázquez Poletti por concederme una entrevista. A Eugeni Graugès por dejarnos visitar el laboratorio de física moderna de la UB y dejarnos hacer la práctica con el generador de rayos X.

## 9. Referencias

### 9.1. Análisis numérico

WIKIPEDIA.Numerical analysis.Varios autores (online).(Fecha de consulta: 16-9-2018)

[https://en.wikipedia.org/wiki/Numerical\\_analysis](https://en.wikipedia.org/wiki/Numerical_analysis)

WIKIPEDIA.Gaussian elimination.Varios autores (online).(Fecha de consulta: 16-9-2018)

[https://en.wikipedia.org/wiki/Gaussian\\_elimination](https://en.wikipedia.org/wiki/Gaussian_elimination)

WIKIPEDIA.QR decomposition.Varios autores (online).( Fecha de consulta : 16-9-2018)

[https://en.wikipedia.org/wiki/QR\\_decomposition](https://en.wikipedia.org/wiki/QR_decomposition)

WIKIPEDIA.Round-off error.Varios autores (online).(Fecha de consulta: 17-9-2018)

[https://en.wikipedia.org/wiki/Round-off\\_error](https://en.wikipedia.org/wiki/Round-off_error)

WIKIPEDIA.Truncation error.Varios autores (online).(Fecha de consulta 17-9-2018)

[https://en.wikipedia.org/wiki/Truncation\\_error](https://en.wikipedia.org/wiki/Truncation_error)

WIKIPEDIA.Discretization error.Varios autores (online).(Fecha de consulta 17-9-2018)

[https://en.wikipedia.org/wiki/Discretization\\_error](https://en.wikipedia.org/wiki/Discretization_error)

WIKIPEDIA.Numerical stability.Varios autores (online).(Fecha de consulta 17-9-2018)

[https://en.wikipedia.org/wiki/Numerical\\_stability](https://en.wikipedia.org/wiki/Numerical_stability)

### 9.2. Análisis numérico aplicado a la ciencia

Prezi.Historia del análisis numérico.Sahori Morales(online).(Fecha de consulta 18-11-2018)

[https://prezi.com/w\\_s8iduerssc/historia-del-analisis-numerico/](https://prezi.com/w_s8iduerssc/historia-del-analisis-numerico/)

Viva-read.Analisis numérico.Varios autores (online).(Fecha de consulta 18-11-2018)

<http://es.viva-read.com/article/anlisis-numrico-computacional>

Multilanguage documents.Historia de los métodos numéricos .Oscar Hernández Mondragon (online). (Fecha de consulta 18-11-2018)

<https://vdocuments.site/documents/historia-de-los-metodos-numericos.html>

WIKIPEDIA.Computación en la nube.Varios autores (online).(Fecha de consulta 18-11-2018)

[https://es.wikipedia.org/wiki/Computación\\_en\\_la\\_nube](https://es.wikipedia.org/wiki/Computación_en_la_nube)

Ticbeat.Qué es cloudcomputing.Varios autores (online).(Fecha de consulta 18-11-2018)

<http://www.ticbeat.com/cloud/que-es-cloud-computing-definicion-concepto-para-neofitos/>

WIKIPEDIA.Computacion grid.Varios autores (online).(Fecha de consulta 18-11-2018)

[https://es.wikipedia.org/wiki/Computación\\_grid](https://es.wikipedia.org/wiki/Computación_grid)

Seti.Varios autores (online).(Fecha de consulta 18-11-2018)

<https://seti.org/>

einsteinathome.Varios autores (online).(Fecha de consulta 18-11-2018)

<https://einsteinathome.org/es/home>

WIKIPEDIA.Einstein@Home.Varios autores (online).(Fecha de consulta 18-11-2018)

<https://es.wikipedia.org/wiki/Einstein@Home>

WIKIPEDIA.SETI.Varios autores (online).(Fecha de consulta 18-11-2018)

<https://es.wikipedia.org/wiki/SETI>

Climatepredictio.Varios autores (online).(Fecha de consulta 18-11-2018)

<https://www.climateprediction.net/>

WIKIPEDIA.Climate prediction.Varios autores (online).(Fecha de consulta 18-11-2018)

<https://en.wikipedia.org/wiki/Climateprediction.net>

### 9.3. Octave

BARTH T.J.; y otros. *Programming for Computations – Matlab/Octave*.Eidelberg, 2010.

GNU. Octave. Varios autores (online).(Fecha de consulta 24-9-2018)

<https://www.gnu.org/software/octave/>

WIKIPEDIA.GNU octave.Varios autores (online).(Fecha de consulta 24-9-2018)

[https://es.wikipedia.org/wiki/GNU\\_Octave](https://es.wikipedia.org/wiki/GNU_Octave)

## 9.4. Analisis de mis scripts

BARTH T.J.; y otros. *Programming for Computations – Matlab/Octave*.Eidelberg, 2010.

## 9.5. Barcelona Supercomputing Center

BSc.Varios autores (presentación).(Fecha de consulta 21-10-2018)

<https://www.bsc.es/>

WIKIPEDIA.Centro Nacional de Supercomputación.Varios autores (online).

(Fecha de consulta 21-10-2018)

[https://es.wikipedia.org/wiki/Centro\\_Nacional\\_de\\_Supercomputación](https://es.wikipedia.org/wiki/Centro_Nacional_de_Supercomputación)

SUSE. Release Notes. Varios autores (online).(Fecha de consulta 16-11-2018)

[https://www.suse.com/releasenotes/x86\\_64/SUSE-SLES/12/](https://www.suse.com/releasenotes/x86_64/SUSE-SLES/12/)

xCAT. Documentation. Varios autores (online).(Fecha de consulta 16-11-2018)

<https://xcat.org/>

Intel software.Información sobre software.Varios autores (online).

(Fecha de consulta 16-11-2018)

<https://software.intel.com/en-us/parallel-studio-xe>

WIKIPEDIA.GNU compiler Collection.Varios autores (online).(Fecha de consulta 16-11-2018)

[https://es.wikipedia.org/wiki/GNU\\_Compiler\\_Collection](https://es.wikipedia.org/wiki/GNU_Compiler_Collection)

Open-mpi.Varios autores (online).(Fecha de consulta 16-11-2018)

<https://www.open-mpi.org/>

WIKIPEDIA.Open MPI.Varios autores (online).(Fecha de consulta 16-11-2018)

[https://en.wikipedia.org/wiki/Open\\_MPI](https://en.wikipedia.org/wiki/Open_MPI)

Bsc.Información sobre el BSC.Varios autores (online).(Fecha de consulta 21-10-2018)

<https://www.bsc.es/>

WIKIPEDIA.Mare Nostrum.Varios autores (online).(Fecha de consulta 21-10-2018)

<https://es.wikipedia.org/wiki/MareNostrum>

## 9.6. Entrevista José Luis Vazquez-Poletti

Dsa-research.Biografía.Varios autores (online).(Fecha de consulta 16-11-2018)

<http://www.dsa-research.org/doku.php?id=people:poletti>

Universidad Complutense de Madrid.Biografía.Varios autores (online). (Fecha de consulta 16-11-2018)

<https://www.ucm.es/directorio?id=19623>

## 9.7. Creación de un servidor local Wordpress

Gianoliveira.Creación página web.Varios autores (online).(Fecha de consulta 29-7-2018)

<https://www.gianoliveira.com/como-instalar-un-servidor-local.html>

Gianoliveira.Creación wordpress.Varios autores (online).(Fecha de consulta 29-7-2018)

<https://www.gianoliveira.com/curso-wordpress/como-instalar-wordpress-paso-a-paso/>

## 9.8. Práctica en la Universidad de Barcelona

Monografías.Como funcionan las ondas electromagnéticas .Varios autores (online). (Fecha de consulta 20-4-2018)

<https://www.monografias.com/docs/Como-Funcionan-Las-Ondas-Electromagneticas-F3SBTFC8G2Z>

WIKIPEDIA.Algoritmos de agrupamiento.Varios autores (online). (Fecha de consulta 25-12-2018)

[https://es.wikipedia.org/wiki/Algoritmo\\_de\\_agrupamiento](https://es.wikipedia.org/wiki/Algoritmo_de_agrupamiento)

Cisco. Learningnetwork .Varios autores (online).(Fecha de consulta 19-12-2018)

<https://learningnetwork.cisco.com/thread/24014>

## 9.9. Parte teórica común

AGUSTIÑO, Mario. (2019): “Els fonaments de la física moderna”. *Radiació, estructura i interaccions de la matèria*.

VALLÈS, Carles. (2019): “Detector de partícules Timepix”. *Radiació ionitzant i barreres de radiació*

GALVE, Guillermo. (2019): “Física Nuclear”.

# ANEXOS

---

# 10. Análisis del software PixetPro

## Introducción

El PIXET Pro es un software multiplataforma desarrollado en la empresa ADVACAM. Es un paquete de software para el control de adquisición de datos para los chips de la familia Medipix. Es compatible con la mayoría de los Medipix disponibles basados en dispositivos (Medipix2, Medipix3, Timepix, Medipix Quad, Timepix Quad, Widepix, Timepix3) y con las interfaces de lectura comúnmente utilizadas (MiniPIX, FitPIX, ModuPIX, WidePIX, RasPIX, etc)

En particular nosotros lo hemos utilizado sobre un chip Timepix con interfaz MiniPIX.

Pixet proporciona muchas herramientas para la optimización de los parámetros del detector, procesamiento de datos, correcciones de imágenes y scripting en Python con resaltado de sintaxis.

Pixet está escrito en lenguaje C++ y utiliza bibliotecas Qt multiplataforma.

## Aplicaciones

Nosotros hemos usado el software para:

- Configurar el detector.
- Medir y leer los datos que nos proporciona el detector Medipix.
- Visualizar y modificar los datos.
- Guardar los datos medidos en diferentes tipos de formatos (ASCII,PNG,pixel log,...).
- Aplicar scripts como Flat-field para mejorar las imágenes.
- Aplicar herramientas para realizar un diagnóstico de la imagen.

## Ventana principal



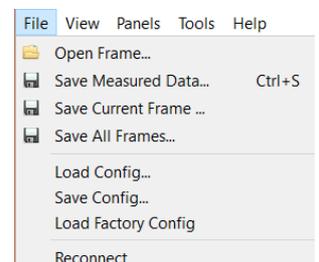
Esta ventana consta de 6 zonas bien diferenciadas:

- La barra de menús: con diferentes acciones y comandos.
- La barra de herramientas: con botones con las principales acciones.
- La barra de dispositivos: con los dispositivos conectados
- La barra de estado: con información de la medición.
- La barra de paneles: con diferentes paneles de configuración y muestra de datos.
- La imagen detectada: con una barra de estado y las escalas de energía.

### La barra de menús

Esta ventana tiene una barra de menús superior con diferentes opciones:

La opción de **File** nos permite abrir el archivo que queremos analizar o modificar. Dentro de esta encontramos la opción de diferentes tipos de guardado dependiendo de lo que queramos guardar. También encontraremos la opción de cargar la configuración usada en archivos anteriores.



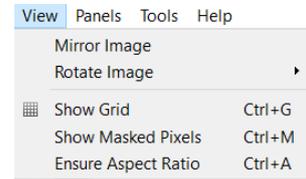
El menú **View** nos permitirá cambiar la manera en que vemos la imagen. Otras opciones son:

**-Mirror image** nos permitirá superponer una imagen igual a la que tenemos.

-**Rotate image** nos permite girar la imagen.

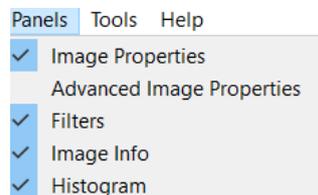
-**Show Grid** divide todos los píxeles rodeándolos con una línea gris.

-**Show Masked Pixels** nos permite visualizar los Píxeles que están enmascarados, es decir, los que a simple vista no se ven o se eliminan para una mejor calidad de imagen.



-**Ensure Aspect Ratio** nos permite mostrar la imagen con la ratio anchura y altura fijas.

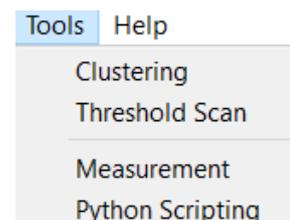
El menú **Panels** permite elegir los paneles que deseamos que se vean en nuestra ventana principal. Estos paneles se explican en la siguiente sección.



El menú **Tools** nos permite usar algunas herramientas para mejorar o modificar nuestra imagen.

**Clustering** nos permitirá agrupar todos los píxeles ya sea por similitud o distancia, consiguiendo que la imagen se vea más suave o homogénea.

**Threshold Scan** te permite detectar alguna detección fuera de lugar.



**Measurement** mide la cantidad de píxeles ya sea en número o por cantidad de energía.

**Python Scripting** modifica el documento pero de una manera más profesional ya que lo modificas directamente en el script (código) de la imagen.

Por último el menú **Help** nos proporciona ayuda para poder usar el programa o resolver alguna duda.

## La barra de dispositivos

En la parte izquierda de la ventana principal encontramos el/los detectores que estamos usando en ese momento. El color cambiará dependiendo de si está enviando información o no.

En mi caso sale de esta manera porque no tengo el detector conectado.

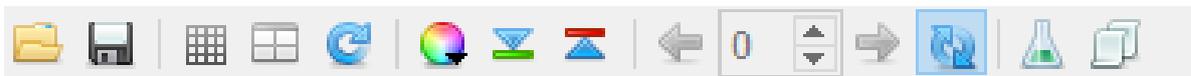


En caso de tener varios chips conectados aparecen los diferentes detectores cada uno con un identificador y un icono con el tipo de detector. El detector activo, se muestra en color mientras que los inactivos se muestran sin resaltar.



## Barra de herramientas

Debajo de la barra de opciones superior encontramos la **barra de herramientas** que también nos permitirá modificar la imagen.



La primera y segunda opción nos permiten abrir y guardar la imagen actual.

La tercera y cuarta nos permiten mostrar la rejilla de píxeles y mostrar el número de serie del chip. La quinta nos permite girar la imagen.

La sexta nos permite cambiar el tipo de color con los que se ve la imagen. La séptima y octava nos permiten modificar el rango de colores según nuestras necesidades.

La novena nos permite pasar de frame en frame o seleccionar un frame en concreto.

La décima nos permite activar o desactivar la opción de auto refrescar.

La undécima nos permite editar la configuración de los píxeles. La duodécima nos permite mostrar o no el icono en el menú de PixetPro

## La barra de estado

En la parte inferior encontraremos la **barra de estado**:



Primero encontramos **Messages** esto nos indica el número de mensajes que nos envía el propio programa.

A continuación encontraremos el número de frames (**FPS**) que se han medido por segundo. Después encontramos el tiempo (**T**) en el que se ha medido.

Más adelante encontramos el contador que nos indicará cuántas repeticiones de la misma imagen se han realizado (**R**). Después encontramos el marcador (**A**) que nos indica cuantas imágenes se han podido medir en total.

También encontraremos el icono del estado (Verde-Ok, Amarillo-advertencia, Rojo-error).

Por último encontraremos la barra de progreso.

## Image Properties:

Nos permite ajustar el valor de energía del rango visible de los fotogramas que deseamos ver. Esto nos permitirá resaltar algunos colores y opacar otros.



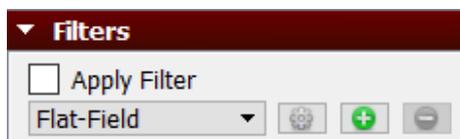
Estos valores van desde el valor mínimo (Min) hasta el valor máximo (Max) y se corresponden con la escala de colores que se haya seleccionado. Por ejemplo, si se trabaja en la escala de grises, los más bajos son los que corresponden al negro y a medida que vamos subiendo aparecerán otros, que se juntarán hasta llegar al blanco. Así, si definimos una matriz 3x3 entre los valores 1 (min) y 6 (max) todos los valores por de 1 o menos se verán negros, todos los valores de 6 o más se verán blancos y los valores intermedios tomarán diferentes tonos de gris.

5	1	0
3	4	1
5	6	2

Existe la opción de **auto range** que ajustará automáticamente estos valores para que la imagen cubra todo el rango.

## Filters

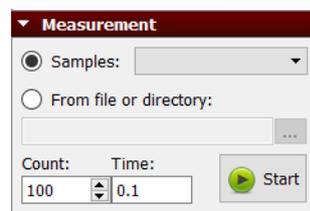
Este panel nos permite aplicar filtros para que la imagen se vea de la mejor manera posible utilizando varios métodos y scripts. Un análisis del script flat-field se encuentra disponible en la parte práctica del TR.



## Measurement

Este panel nos permite elegir desde donde vamos a adquirir los datos para generar la imagen. Las opciones disponibles son desde el propio detector, instantáneamente o desde un documento guardado.

También nos va a permitir seleccionar los parámetros de tiempo y de cantidad de las veces que vamos a realizar la prueba.



Igualmente nos permite seleccionar el tipo de medida, pudiendo seleccionar por frames (imágenes individuales), integral (suma de imágenes) o por inyección de pulsos (solo para comprobación del chip).

Por último nos va a permitir ejecutar el programa e iniciar el proceso de medida.

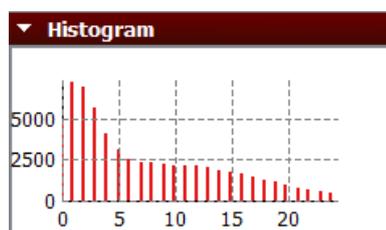
## Image Info

Este panel nos muestra información estadística de la imagen, la posición del cursor... La información que se muestra en este panel es la que corresponde a la imagen mostrada por pantalla, por lo que si en pantalla solo se muestran 4 píxels por haber hecho un zoom, la información hará referencia a la estadística calculada con esos 4 píxels.

Image Info	
[X,Y]:	[170, 94]
Count:	7
Min:	0
Max:	112
Pixel Count:	60922
Total:	585153
Mean:	8.92873

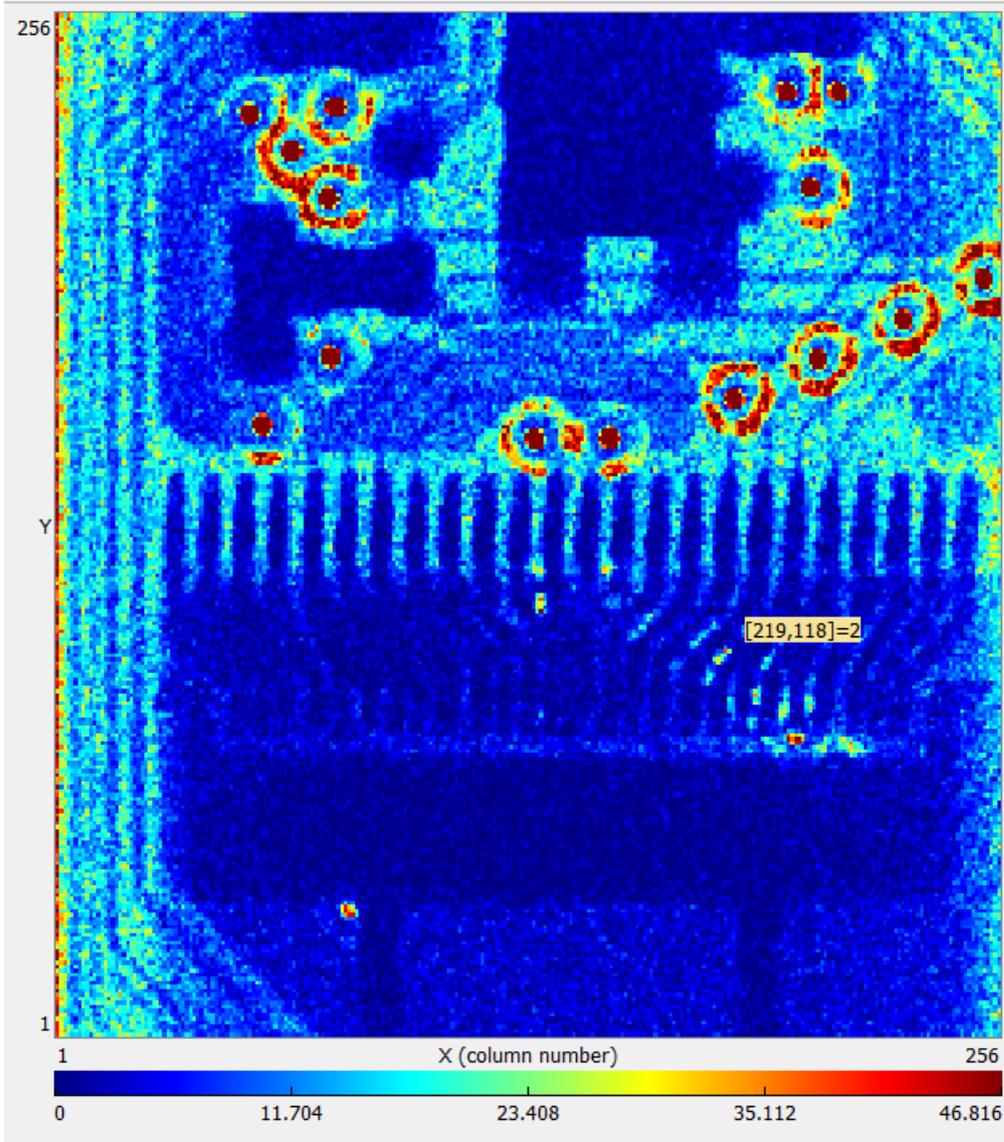
## Histogram

El último panel nos crea un histograma de la imagen o imágenes con las que estamos trabajando. Puede modificarse la escala y también puede exportarse para analizarse en una hoja de cálculo externa.



## Panel de imagen

Por último encontramos en el centro de todo la parte donde veremos la imagen y sus medidas y parámetros.



En este caso se muestra una de las pruebas que se hicieron en la universidad de Barcelona con el detector. En concreto esta muestra una memoria SD vista con rayos X. Se ha seleccionado como escala de colores *Jet* y una escala de energías entre 0 i 46,82 keV

# 11. Pràctica Detecció de partícules

Esta práctica se llevó a cabo en el laboratorio de física del colegio en mayo del 2018. La hicimos los cuatro alumnos que hemos hecho trabajos de investigación usando el detector TimePix, para aprender a utilizarlo y aprender las propiedades de la radiación. A continuación se presenta el guion de dicha práctica, tal y como nos fue proporcionado por el tutor del TR.

## Objectius

- Iniciar-se en l'ús d'un detector de partícules TimePix i del programari PixetPro.
- Identificar els diferents tipus de radiació pel rastre que deixen en el detector, i relacionar aquests rastres amb les característiques que presenten les partícules que el formen.
- Mesurar l'energia cinètica de les partícules absorbides al detector.
- Comprovar la importància de la desviació relativista en el càlcul de la velocitat de les partícules per a la radiació beta.
- Determinar les característiques dels fotons constituents de la radiació gamma.

## Introducció teòrica

### Física nuclear. Les radiacions. Energia relativista.

La radioactivitat és un procés natural i espontani per al qual els àtoms inestables d'un element emeten o irradien l'excés d'energia del seu nucli i, així, canvien (o decauen) a àtoms d'un element diferent o un estat d'energia menor del mateix element. Els àtoms aconseguen ser inestables variant l'energia dels seus electrons, dels seus nucleons o variant d'isòtop. Aleshores, els elements o cossos anomenats radioactius, emeten radiacions que tenen la propietat d'impressionar plaques fotogràfiques, ionitzar gasos, produir fluorescència...

Els nuclis dels elements radioactius primaris es van formar, juntament amb nuclis estables d'elements, per síntesi nuclear en els nuclis d'estrelles o per nucleogènesi durant explosions de supernoves. Si aquests nuclis tenien una vida mitjana curta (en comparació amb l'edat de la Terra), van decaure i ja no estan presents en el medi ambient. Solament aquells amb una vida mitjana de  $10^8$  anys com a mínim es poden trobar al nostre voltant. Els radionúclids primaris més importants són el potassi  $^{40}\text{K}$  (vida mitjana d' $1,26 \times 10^9$  anys), tori  $^{232}\text{Th}$  (vida mitjana  $1,40 \times 10^{10}$  anys),  $^{238}\text{U}$  d'urani (vida mitjana de  $4,47 \times 10^9$  anys) i  $^{235}\text{U}$  d'urani (vida mitjana  $7,04 \times 10^8$  anys). Tori i urani formen una sèrie de decaïment característica, al contrari que el potassi.

Les radiacions que emeten els nuclis radioactius es divideixen en 3 tipus ben diferenciats amb naturalesa i comportament ben diferenciats:

- radiació alfa ( $\alpha$ ): formada per nuclis d'heli. En emetre una partícula alfa, el nucli perd dos protons i dos neutrons. Es tracta d'una partícula relativament pesada i que

interacciona fortament amb la matèria. És molt ionitzant i, per tant, poc penetrant.

- radiació beta ( $\beta$ ): formada per electrons ( $\beta^-$ ) o positrons ( $\beta^+$ ) emesos pel nucli. En emetre una partícula beta, un neutró del nucli es converteix en un protó, o un protó del nucli es converteix en un neutró. Es tracta d'una partícula molt més lleugera que l'anterior, que interacciona menys amb la matèria i per tant té molta més penetració.

- radiació gamma ( $\gamma$ ): formada per fotons (ones EM) d'alta energia. En emetre fotons el nucli no canvia la seva naturalesa, sinó que només s'allibera d'energia sobrant. Es tracta de la partícula més lleugera i per tant interacciona molt menys amb la matèria, cosa que la fa molt més penetrant.

**Com ja saps, les radiacions ionitzants són molt perilloses perquè poden trencar els enllaços de l'ADN i produir mutacions a nivell cel·lular. Les mostres amb què treballarem nosaltres són de baixa activitat i no són perilloses. Malgrat tot, no les toquis amb les mans. Si ho fas, ves a rentar-te-les sense tocar-te la boca ni els ulls.**

Per la seva naturalesa, l'estudi d'aquestes partícules ha de tenir en compte el seu comportament quàntic i relativista. Així la radiació gamma es comporta com a partícules (fotons) quan interaccionen amb els àtoms de Si del sensor. Aquest fotons tenen una quantitat de moviment donada per  $p = E/c$ , on  $c = 2,99792 \cdot 10^8$  m/s és la velocitat de la llum.

Igualment, les partícules  $\beta$  es mouen a una fracció significativa de la velocitat de la llum i per tant l'expressió clàssica de la energia cinètica,  $E_k = \frac{1}{2}mv^2$ , no és vàlida i cal emprar les expressions de la relativitat especial.

Així, l'energia de les partícules ve donada per la relació d'Einstein  $E = E_0 + E_k$  on  $E$  és l'energia total de la partícula,  $E_0$  l'energia en repòs i  $E_k$  l'energia cinètica. Tenint en compte l'equivalència massa-energia,  $E = mc^2$ , i substituint

$$E = \frac{E_0}{\sqrt{1 - \frac{v^2}{c^2}}} \quad (1)$$

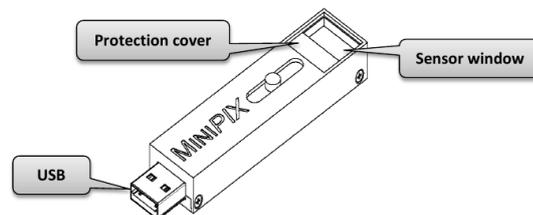
podem expressar la velocitat d'una partícula alfa o beta com

$$v = c \sqrt{1 - \frac{1}{\left(1 + \frac{E_k}{E_0}\right)^2}} \quad (2)$$

## El detector

MINIPIX és una solució de càmera de radiació de detector Timepix amb recompte de partícules individuals (o seguiment de partícules), miniaturitzada i de baixa potència, desenvolupat al CERN. El sistema estàndard MINIPIX incorpora un detector Single Timepix (256 x 256 píxels amb un camp de 55  $\mu\text{m}$  cada un) amb un sensor de silici estàndard de 300 micres de gruix. Utilitza la interfície USB 2.0. i és capaç de llegir fins a 30 fotogrames per segon (amb un temps d'exposició d'1 ms). El detector Timepix és sensible a l'energia, cosa que aporta una nova dimensió a les imatges radiogràfiques. Un sensor com aquest (entre altres dispositius) és el que utilitzen els astronautes de la NASA a la ISS per a detectar radiació còsmica.

<https://home.cern/about/updates/2012/08/timepix-detectors-track-cosmic-radiation-iss>



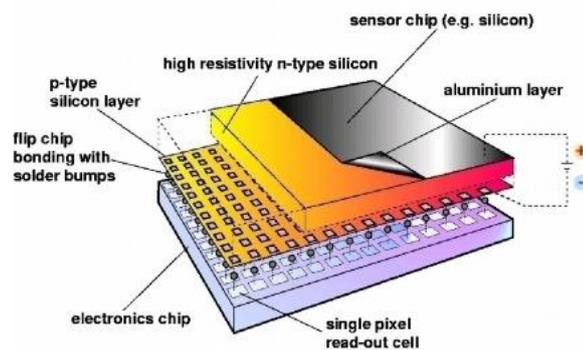
### IMPORTANT:

**No tocar la superfície de la finestra del sensor ni amb els dits ni amb qualsevol altre objecte. La làmina de silici de 0,3 mm és molt delicada.**

**Mantenir allunyat de l'aigua i la humitat.**

**No desmuntar el dispositiu.**

La detecció de radiació ionitzant es produeix al xip detector, dissenyat com un díode semiconductor pla de 14mmx14mm i un gruix de 300 micròmetres. El díode està connectat en polarització inversa, l'usuari pot triar el voltatge de la polarització (anomenat bias) fins a 100 V. Això crea una zona lliure de càrregues en la junció P-N i el díode només transmet una petita quantitat de corrent (anomenada dark current) que pot ignorar-se triant un valor llindar (treshold) apropiat.



Quan la radiació ionitzant impacta sobre el Si, genera parells de electró-forat (l'energia mínima necessària és de 3,62 eV) que són atrets pels elèctrodes, produint un pols de corrent detectable. Com el detector no és un únic díode si no que la superfície està configurada com una graella de 65536 segments (256x256 díodes), el detector és capaç de mostrar en quins píxels s'ha produït el senyal. Segons el tipus de partícula que formi la radiació (i segons els paràmetres de configuració del detector), la interacció amb els àtoms dels díodes de silici serà diferent i per tant el senyal produït també, podent distingir el tipus de partícula a partir del senyal que

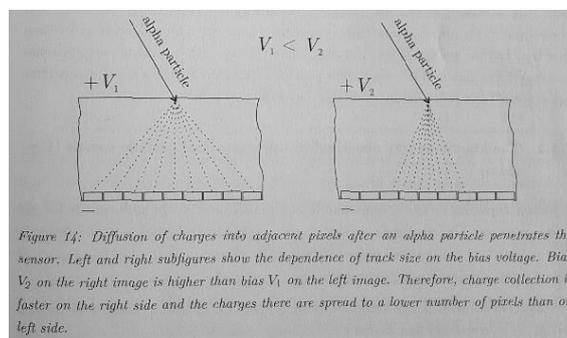


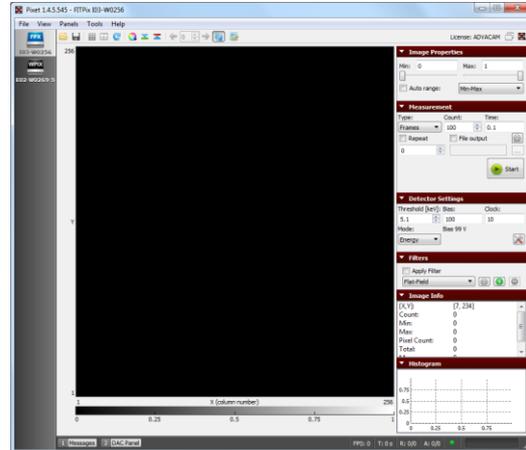
Figure 14: Diffusion of charges into adjacent pixels after an alpha particle penetrates the sensor. Left and right subfigures show the dependence of track size on the bias voltage. Bias  $V_2$  on the right image is higher than bias  $V_1$  on the left image. Therefore, charge collection is faster on the right side and the charges there are spread to a lower number of pixels than on left side.

produeix en el sensor.

A més a més, el processament del senyal ens permet treballar en mode comptador, per tal de comptar el nombre de díodes activats per la partícula, en mode temps per tal de determinar la durada de la interacció, o en mode energia, que ens permet mesurar l'energia de la partícula irradiant si el detector està calibrat convenientment.

## El programari

El *PIXET Pro* és un programari multiplataforma desenvolupat a l'empresa ADVACAM. És un paquet de programari per al control d'adquisició de dades per a xips de la família Medipix. Pixet proporciona moltes eines per a l'optimització de paràmetres del detector, processament de dades, correccions d'imatges i scripting.



### La finestra principal

La finestra principal del programari consisteix en la barra lateral del dispositiu, el quadre (imatge) on es visualitzen les dades mesurades, el tauler de control al costat dret, la barra d'eines, la barra de menú i el panell d'estat abaix. Els panells al costat dret es poden minimitzar fent clic a la fletxa cap avall del panell corresponent o es pot amagar / mostrar a través dels panells de menú.

### Barra de dispositius

Mostra els dispositius connectats. Si no apareix el dispositiu en colors, aviseu al professor.

### Quadre

El quadre mostra les dades mesurades (quadres)

Zoom: Prement el botó esquerre i arrossegant el ratolí per crear un marc rectangular es farà zoom a la zona emmarcada. Aquesta zona pot ajustar-se per les barres de scroll o movent la imatge amb el botó dret del ratolí.

Fent doble clic, el zoom es desfà

### Image Properties

Permet modificar les propietats de la imatge que es mostrarà al quadre.

### Measurement

Permet determinar els paràmetres de la mesura, com nombre de quadres, temps, ...

També permet escollir el mode: *frames*, que mesura un quadre cada vegada, i *integral*, que acumula cada quadre sobre els anteriors.

El botó *Start* Inicia la mesura i el botó *Stop* l'atura.

### Detector Settings

Aquest panell permet modificar els paràmetres del dispositiu.

### *Filters*

Permet aplicar diferents filtres i correccions de la visualització de les dades mesurades, sense modificar les pròpies dades.

### *Image Info*

Mostra informació estadística de la imatge actual en el quadre.

### *Histogram*

Mostra l'histograma de valors del pixel per al quadre actual.

### *Toolbar*

Té alguns botons força útils. D'esquerra a dreta

Open Frame - Save Measured Data - Show Grid - Rotate Image - Color map - Under warning - Over warning - Previous frame – Frame - Next frame - Auto update - Edit Pixel Configuration – Measurements - Python scripting



## **Realització de la pràctica**

### **Material**

- Detector TimePix.
- Ordinador.
- Programari Pixet Pro.
- Full de càlcul.
- Mostra radioactiva.
- Guants de plàstic.
- Cullereta.
- Pines.

### **Muntatge de l'experiència**

En aquesta experiència utilitzarem el detector en dos modes diferents i per tant caldrà fer dues mesures amb parametrizacions diferents del programa. Quan acabis la primera, completa la secció adquisició i enregistrament de les dades.

Connecta el detector a un port USB de l'ordinador, sense obrir la protecció. En primer lloc ens assegurarem que el programa connecta amb el detector, seleccionant el botó Show Chip Numbers de la Toolbar. Al quadre ha d'aparèixer en gran, G08-W0253. Si no és així, avisar al professor abans de continuar.

### *Determinació de les trajectòries*

Un cop estiguis segur de que el detector es comunica correctament amb el programa, fixa la ràtio de la imatge al menú *View* i selecciona el color Jet White a la *Toolbar*. Val la pena que premis el botó *grid* per a una millor visualització dels píxels del sensor.

Per a fer aquestes mesures, cal que configureu el detector amb la següent parametrització:

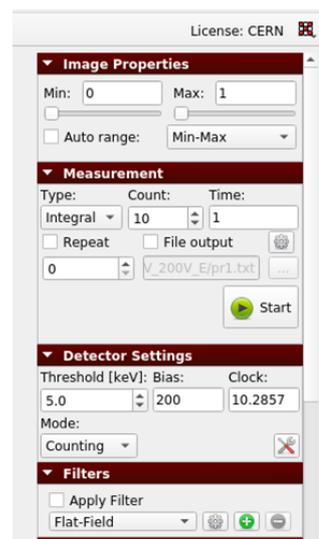
Obriu la porta de protecció del detector.

Apropeu la mostra radioactiva al sensor i premeu el botó Start per iniciar la captura de dades. Com hem seleccionat el mode integral, els quadres que detecta es van acumulant. A mesura que es van obtenint dades veureu 3 tipus de trajectòries, una per a cada tipus de radiació.

Veureu trajectòries grans i arrodonides, anomenades Blobs (gotes) causades per les partícules alfa. Més comunes són unes trajectòries llargues i sovint corbades, anomenades Worms (cucs), produïdes per la radiació beta d'alta energia. Finalment, la radiació beta de baixa energia i la radiació gamma produeixen trajectòries petites i curtes, normalment de un (o pocs) píxels.

No és possible distingir quin tipus de radiació ha produït aquestes trajectòries amb el nostre detector.

**Quan s'aturi el procés d'obtenció de dades, tanqueu el protector.**



### *Energia cinètica absorbida pel sensor i velocitat de les partícules en els models clàssic i relativista.*

A continuació anem a treballar en mode energia. Una partícula detectada en el xip pot ser absorbida al sensor i depositar-hi tota la seva energia cinètica (com passa en el cas de les partícules alfa i les beta de baixa energia) o travessa tot el detector dipositant només una part de la seva energia (cas de les partícules beta d'alta energia).

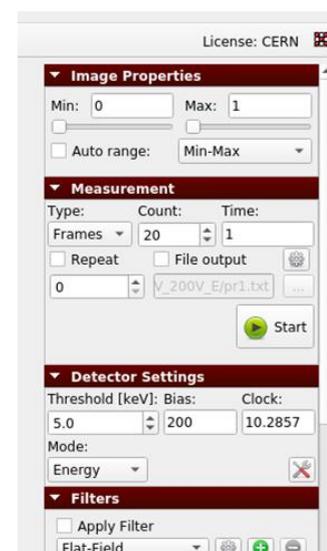
Parametritzeu el detector com a la imatge:

Pot ser interessant seleccionar l'opció Auto range, que ajustarà el color de l'energia detectada a cada pixel mostrat a la imatge.

Obriu la porta de protecció del detector.

Apropeu la mostra radioactiva al sensor i premeu el botó Start per iniciar la captura de dades. Com hem seleccionat el mode Frames, cada quadre detectat substitueix el quadre anterior.

**Quan el procés d'obtenció de dades finalitzi, tanqueu el protector del xip.**



Si observeu el panell *Image Info*, veureu una sèrie de valors interessants, com el valor màxim d'energia detectada i el total d'energia que s'ha dipositat en les partícules detectades al quadre que es mostra. Aquest és el valor d'energia cinètica que tenia la partícula al entrar al detector.

Aneu passant pels diferents quadres amb el selector de la Toolbar i feu zoom sobre diferents trajectòries de les partícules alfa, beta i gamma de manera que apareixin

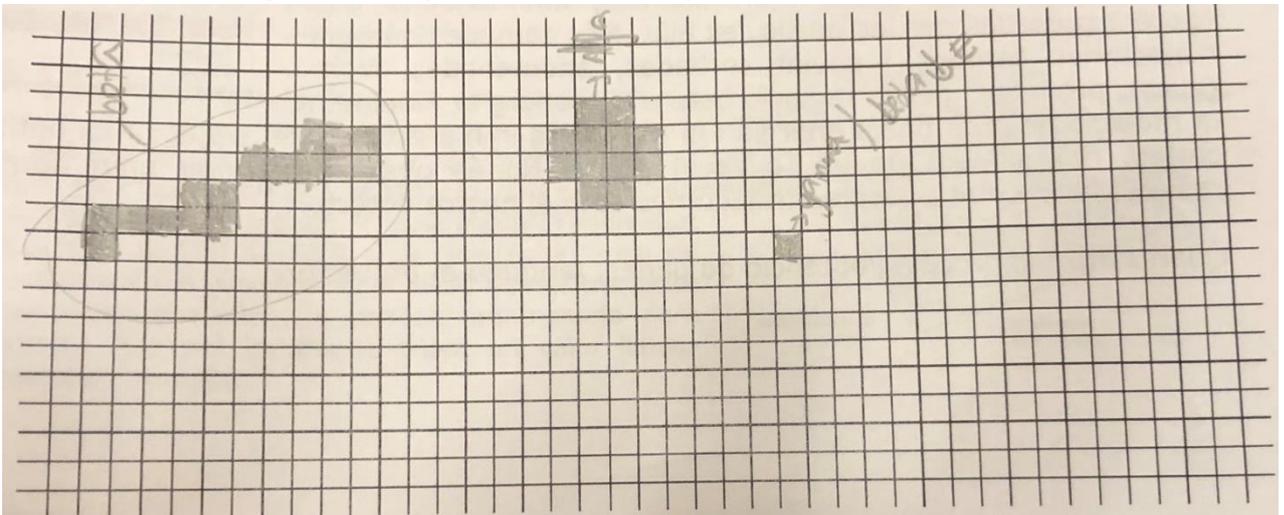
soles a la imatge. Anoteu l'energia que apareix al panell Imatge Info per a cada una de les partícules. Aquest valor està calibrat en keV.

**Tanqueu de nou la protecció del sensor.**

Adquisició i enregistrament de les dades

**1 Determinació de trajectòries.**

1.1 Dibuixa a continuació una de les trajectòries d'una partícula alfa, una partícula beta i una partícula gamma i etiqueta cada un dels tipus.



1.2 Busca a internet la massa en repòs i calcula l'energia en repòs que tenen les partícules alfa i beta.

$$E_0 = 6,68 * 10^{-27} * (3,00 * 10^8)^2 = 6,012 * 10^{-10} J = 3,76 * 10^9 ev$$

$$E_\beta = 9,11 * 10^{-31} * (3,00 * 10^8)^2 = 8,199 * 10^{-14} = 5,12 * 10^5 ev$$

1.3 A partir de les característiques que tenen les partícules de cada tipus de radiació, fes hipòtesis de com aquestes partícules interaccionen amb els àtoms de silici i són detectades pel sensor, donant lloc als diferents tipus de trajectòries (aprox. 50 paraules)

En el cas de les partícules alfa estan formades per nuclis d'heli. Són molt grans i pesats i al trobar-se la placa de silici faran una marca més ampla perquè xocaran amb uns pocs àtoms de silici propers entre sí.

Les partícules beta són electrons i positrons, partícules més petites i viatgen a més velocitat.

Deixen una marca semblant a la d'un raig ja que va xocant amb molts àtoms que la desvien i va recorrent la placa de silici fins que s'acaba la seva energia.

Les partícules gamma són fotons d'alta energia. Aquestes tenen més poder de penetració ja que són molt més petites i no tenen càrrega i per tant quan xoca amb el detector de partícules l'entrevessa deixant una marca molt més petita que les demes radiacions.

## 2 Energia cinètica absorbida pel sensor i velocitat de les partícules alfa en els models clàssic i relativista.

Per a fer els càlculs, crea un full de càlcul a drive de nom PF6\_COGNOM\_NOM i comparteix-lo amb el professor. Crea una taula com la següent. Anota els resultats en aquesta taula. Per a fer els càlculs caldrà fer canvis d'unitats.

<b><math>E_K</math> (KeV)</b>	8199	6561	4624	12289	7674
<b>v (relat.)</b> ( $ms^{-1}$ )	$1'98 \cdot 10^7$	$1'77 \cdot 10^7$	$1'49 \cdot 10^7$	$2'42 \cdot 10^7$	$1'92 \cdot 10^7$
<b>v (class.)</b> ( $ms^{-1}$ )	$1'98 \cdot 10^7$	$1'77 \cdot 10^7$	$1'49 \cdot 10^7$	$2'98 \cdot 10^7$	$1'92 \cdot 10^7$
<b>v/c (relat.)</b> ( $ms^{-1}$ )	0'066	0'059	0'05	0'081	0'064
<b>v/c (class.)</b> ( $ms^{-1}$ )	0'066	0'059	0'05	0'099	0'064
<b>Desviació</b> (%).	0	0	0	0'231	0

3 Energia cinètica absorbida pel sensor i velocitat de les partícules beta en els models clàssic i relativista.

En una nova pàgina del full de càlcul a drive crea una taula com l'anterior. Anota els resultats en aquesta taula.

<b><math>E_K</math> (KeV)</b>	2660	1884	3106	3107	843
<b>v (relat.)</b> ( $ms^{-1}$ )	$2'96 \cdot 10^8$	$2'56 \cdot 10^7$	$2'08 \cdot 10^7$	$2'08 \cdot 10^7$	$1'72 \cdot 10^7$
<b>v (class.)</b> ( $ms^{-1}$ )	$9'67 \cdot 10^8$	$2'57 \cdot 10^7$	$2'08 \cdot 10^7$	$2'08 \cdot 10^7$	$1'72 \cdot 10^7$
<b>v/c (relat.)</b> ( $ms^{-1}$ )	0'99	0'085	0'069	0'069	0'057
<b>v/c (class.)</b> ( $ms^{-1}$ )	3'22	0'085	0'069	0'069	0'057
<b>Desviació</b> (%).	2'27	0	0	0	0

- 4 Energia cinètica absorbida pel sensor i característiques de les partícules gamma. En una nova pàgina del full de càlcul a drive crea una taula com la següent. Anota els resultats dels càlculs a continuació.
- 4.1 Anota en el full de càlcul, en una nova pàgina, el valor de les energies cinètiques que has mesurat per a 5 partícules gamma.
- 4.2 Calcula el valor de la freqüència del fotó, de la seva longitud d'ona i de la quantitat de moviment que ha cedit el fotó a l'àtom de silici.

<b><math>E_K</math> (KeV)</b>	126	46	73	70	75
<b>f (Hz)</b>	$1'9 \cdot 10^{35}$	$6'94 \cdot 10^{34}$	$1'1 \cdot 10^{35}$	$1'05 \cdot 10^{35}$	$1'13 \cdot 10^{35}$
<b><math>\lambda</math>(m)</b>	$1'58 \cdot 10^{-27}$	$4'32 \cdot 10^{-27}$	$2'73 \cdot 10^{-27}$	$2'83 \cdot 10^{-27}$	$2'65 \cdot 10^{-27}$
<b>p (kg·mm/s)</b>	$4'19 \cdot 10^{61}$	$1'53 \cdot 10^{61}$	$2'43 \cdot 10^{61}$	$2'34 \cdot 10^{61}$	$2'5 \cdot 10^{61}$

## 12. Práctica en la Universidad de Barcelona

El día 5 de marzo asistimos a una práctica en Barcelona. Esta actividad la realizamos en el laboratorio de física moderna de la Facultad de Física de la Universidad de Barcelona, que contaba con multitud de máquinas. Nosotros solo trabajamos con la máquina de rayos X, con la que analizamos la penetración de la radiación en los diferentes objetos expuestos a los rayos X.

Pudimos realizar la práctica gracias al Dr. Rafael Ballabriga ya que hizo posible que nos dejaran la máquina de rayos X.

También nos apoyó el Dr. Eugeni quien nos explicó la manera de protegerse de las diferentes radiaciones mediante el uso de determinados materiales y su grado de penetración.

Cabe destacar la presencia del Dr. Sergio Gómez Fernández doctorado en electrónica y su alumno los cuales nos dieron algunos consejos sobre lo que podemos hacer con el detector y algunas ideas de cómo realizar nuestras prácticas.

El tutor de TR Dani Parcerisas nos llevó hasta el lugar y nos acompañó durante toda la experiencia.

En esta visita utilizamos una máquina de rayos X de la marca Phywe, concretamente el modelo XR 4.0. Básicamente esta máquina se utiliza en diferentes campos como puede ser el físico, biológico, en el campo de la geología, ... En el campo físico, que es el que nos interesa a nosotros, se utiliza para analizar espectros de rayos, para la física de cuerpos sólidos, la difracción de rayos X, la espectroscopia de rayos X, radiología y tomografía computarizada de rayos X.

Primero Rafa modificó la fuente de rayos X añadiendo una lente para focalizar el rayo. A continuación situó la muestra a unos 20 centímetros del foco de ondas electromagnéticas y detrás de la muestra nuestro detector de partículas Timepix.

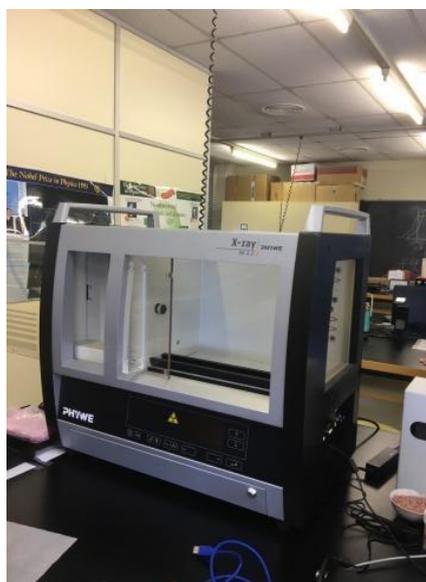
Las ondas electromagnéticas generadas en la fuente de radiación, tras interactuar con la muestra, fueron detectadas por el sensor y convertidas en una señal eléctrica que el ordenador puede procesar y transformar en una imagen.

Como curiosidad, no hubo necesidad de retirar la protección de aluminio del sensor ya que la radiación con la que trabajamos tiene un alto poder de penetración y para ella la cubierta de aluminio era transparente

Comprobamos que no todos los materiales reaccionan de la misma manera.

Los electrones en algunos casos atravesaban el material como si nada y en cambio con otras muestras como el USB observábamos los diferentes circuitos y como las resistencias no dejaban atravesar con facilidad los electrones.

Rafa nos dirigió totalmente la práctica ya que nosotros no sabíamos cómo manipular la máquina de rayos x con sus respectivos complementos, algunos de ellos radiactivos y muy sensibles. Las imágenes que tomamos serán analizadas y personalmente intentaré mejorarlas aumentando la nitidez con el uso del programa Octave.

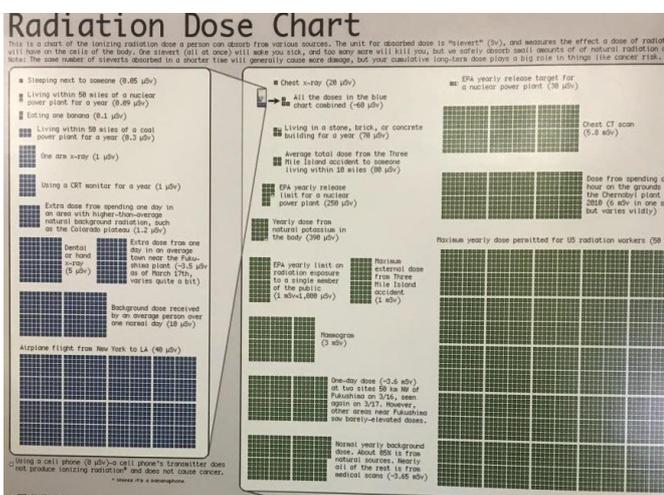


Además, algunas de las imágenes obtenidas se han presentado al concurso de fotografías de física de la Societat Catalana de Física.

Para aumentar la nitidez utilizaré una técnica llamada Flat Field que consiste en capturar la radiación sin muestra, analizar los píxeles de dicha imagen y corregir la imagen con muestra a partir de la información obtenida.

En conclusión, esta práctica me ha parecido muy entretenida a la par que útil para lo que queremos hacer ya que nos ha enseñado cómo actúan las partículas

También y como curiosidad, pudimos consultar una tabla donde muestra la radiación que recibimos cuando viajamos.



## 13. Creación de un servidor de Wordpress

Wordpress es un sistema de gestión de contenidos que permite crear blogs u otros tipos de páginas web. Este sistema funciona mediante un código abierto escrito en PHP lo cual hace muy fácil la creación de webs.

Wordpress surgió gracias a Matt Mullenweg y comenzó siendo una página para colgar las fotografías de su viaje y acabó siendo una plataforma de blogging la cual ha ido evolucionando a un sistema de CMS que permite crear cualquier tipo de sitio web.

El sistema que utilizaremos, Wordpress.org, se basa en 3 directorios:

**Wp-admin** → Donde se encuentran los archivos para el funcionamiento del panel de administración (back-end).

**Wp-content** → Se encuentran los archivos que componen la interfaz del usuario como imágenes, plugins o plantillas (front-end).

**Wp-includes** → Se encuentran los archivos de las aplicaciones básicas y avanzadas para el funcionamiento de wordpress.

Para crear una página web necesitaremos básicamente los Plugins, los temas y los widgets.

Los **plugins** son complementos de software que debemos descargar de la página oficial de wordpress y se usan para mejorar las diferentes áreas de la página web. También pueden ser usados como tutoriales ya que hay de diferentes temáticas como pueden ser la seguridad o las redes sociales ...

En nuestro caso utilizaremos el plugin **polylang** el cual nos permitirá disponer de nuestro blog en varios idiomas.

**Temas:** son las plantillas prediseñadas por wordpress gracias a las cuales podemos modificar la apariencia y el diseño del sitio. Los 3 temas más recomendables para páginas donde se busca crear un negocio son por este orden:

**Widgets:** Básicamente un widget es un programa que te da acceso a funciones que usas de manera frecuente.

En wordpress encontraremos unos 35 widgets en la barra de navegación lo que significa que puedes hacer 35 cosas diferentes

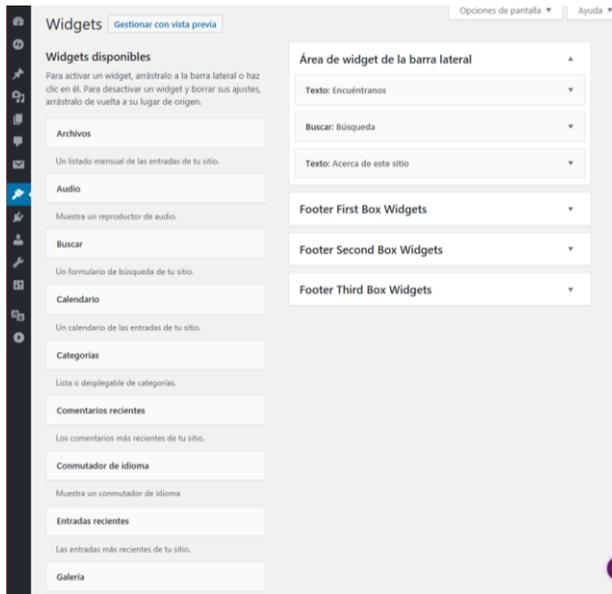


Imagen 8: Centro de control de Wordpress

## 11.1. Instalación

El proceso de instalación consta de 3 partes principales.

En primer lugar habrá que instalar un servidor php. Aunque puede ser instalado manualmente, existen soluciones preprogramadas que facilitan el proceso de instalaciones. En nuestro caso hemos decidido instalar Wampserver, un entorno de desarrollo web de Windows, que permite crear aplicaciones web con apache2, PHP y una base de datos MySQL.

En segundo lugar, será necesario crear la base de datos, cosa que haremos con phpMyAdmin, que viene integrado en Wampserver.

Finalmente instalaremos el paquete de Wordpress.org sobre la base de datos creada.

### 13.1.1. Creación del servidor php

- En primer lugar descargamos Wampserver desde la página oficial ([www.wampserver.com](http://www.wampserver.com)). A pesar de que nuestro servidor tiene más de 4GB de RAM, descargamos la versión de 32 bits para evitar problemas de compatibilidad.

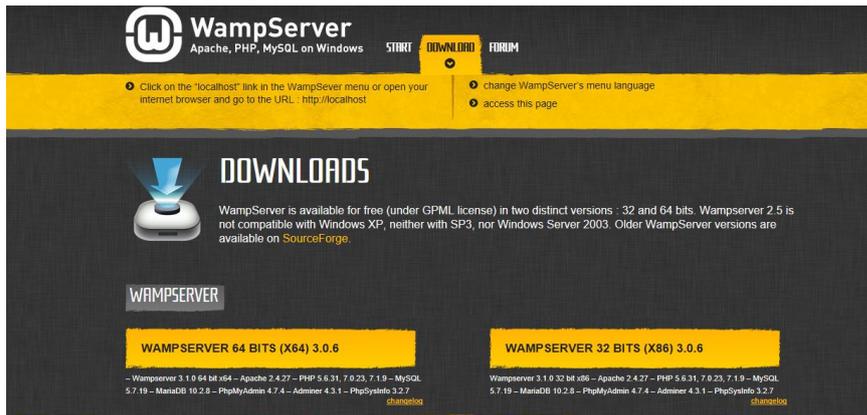


Imagen 9: Página principal de Wamp

Debemos tener cuidado de dejar la ruta de instalación del programa predeterminada que es **C:\wamp\** así como la ruta donde deberemos alojar los datos (en nuestro caso alojaremos el Sistema operativo de wordpress) debe ser también la predeterminada **C:\wamp\www\**. En caso contrario podemos tener problemas después de la instalación.

El mismo programa creará un icono en tu escritorio para que arranques la app desde allí.

El icono variará de color dependiendo si estas conectado totalmente como servidor local (**verde**), parcialmente (**amarillo**) o no estas conectado (**rojo**).

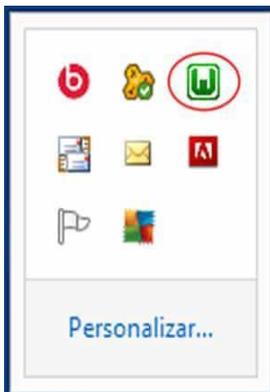


Imagen 10: Icono que nos aparecerá

Cuando abramos la aplicación nos encontraremos con un menú donde si nos fijamos en el apartado de herramientas (tools) podremos ver el enlace a Phpmyadmin es una aplicación que nos permite manejar la creación y administración de nuestra base de datos.

Si nos fijamos en el apartado de nuestros proyectos (Your projects) podremos observarlas diferentes carpetas creadas en **C:\wamp\www\**, es aquí donde

posteriormente deberemos instalar wordpress.

### 13.1.2. Creación de la base de datos

Primero debemos acceder a nuestro servidor local. Esto lo debemos hacer desde el navegador escribiendo: <http://localhost/> y esto nos llevará al menú de wampserver.

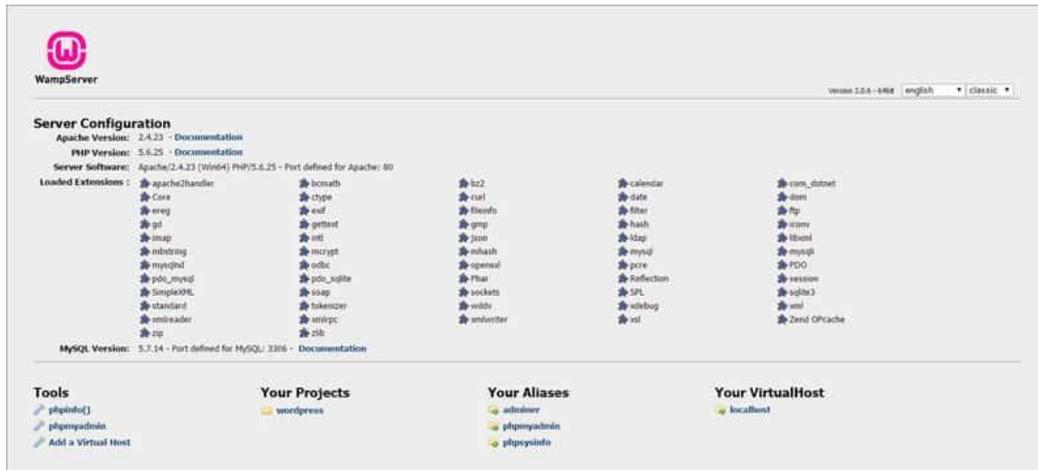


Imagen 11:Menú del Wamp server

Debemos entrar en phpMyAdmin y allí te solicitará un nombre de usuario el cual será **root**.



Ya dentro deberemos crear una base de datos.

Nos dirigiremos a 'crear base de datos' y escribiremos el nombre de la base de datos que queremos crear, en mi caso es Wordpress y le asignaremos un cotejamiento Utf8\_spanish\_ci.



The screenshot shows a web-based interface for managing a MySQL database. At the top, there is a search bar containing 'wordpress' and a dropdown menu set to 'utf8\_spanish\_ci'. A 'Crear' button is visible to the right. Below this is a table with three columns: 'Base de datos', 'Cotejamiento', and 'Acción'. The table lists several databases and their collations, with a 'Total: 5' row at the bottom.

Base de datos	Cotejamiento	Acción
<input type="checkbox"/> information_schema	utf8_general_ci	Seleccionar privilegios
<input type="checkbox"/> mysql	latin1_swedish_ci	Seleccionar privilegios
<input type="checkbox"/> performance_schema	utf8_general_ci	Seleccionar privilegios
<input type="checkbox"/> sys	utf8_general_ci	Seleccionar privilegios
<input type="checkbox"/> wordpress	utf8_spanish_ci	Seleccionar privilegios
<b>Total: 5</b>	latin1_swedish_ci	

Imagen 12: Nuestra base de datos desglosada

### 13.1.3. Instalación de wordpress

Como ya hemos creado la base de datos, nos dirigiremos a nuestra carpeta (C:\wamp\) y en el directorio www crearemos una carpeta llamada wordpress y allí dentro deberemos descomprimir el archivo en su interior.

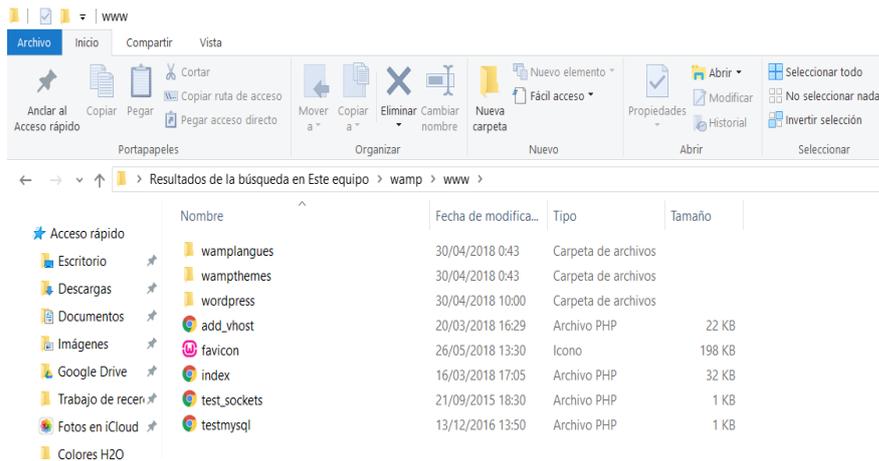


Imagen 13:Lugar donde se encuentran los archivos

Con los archivos descomprimidos dentro de la carpeta introduciremos en el navegador el siguiente enlace <http://localhost/wordpress/>, con esto conseguiremos llegar al menú de instalación de wordpress.



Presionaremos a jvamos a ello ! y seguiremos todos los pasos que nos diga, al final nos debe quedar una cosa como esta:

A continuación deberás introducir los detalles de conexión a tu base de datos. Si no estás seguro de esta información contacta con tu proveedor de alojamiento web.

**Nombre de la base de datos**  El nombre de la base de datos que quieres usar con WordPress.

**Nombre de usuario**  El nombre de usuario de tu base de datos.

**Contraseña**  La contraseña de tu base de datos.

**Servidor de la base de datos**  Deberías recibir esta información de tu proveedor de alojamiento web, si localhost no funciona.

**Prefijo de tabla**  Si quieres ejecutar varias instalaciones de WordPress en una sola base de datos cambia esto.

Imagen 14:Menú de creación de Wordpress

Hacemos clic a enviar y nos llevará a la siguiente pantalla la cual debemos rellenar colocando el nombre de tu sitio.

Como nombre de usuario y contraseña utilizaremos **admin** y en el apartado de correo pondremos nuestro correo

Hola

¡Bienvenido al famoso proceso de instalación de WordPress en cinco minutos! Simplemente completa la información siguiente y estarás a punto de usar la más enriquecedora y potente plataforma de publicación personal del mundo.

**Información necesaria**

Por favor, debes facilitarnos los siguientes datos. No te preocupes, siempre podrás cambiar estos ajustes más tarde.

**Título del sitio**

**Nombre de usuario**   
Los nombres de usuario pueden tener únicamente caracteres alfanuméricos, espacios, guiones bajos, guiones medios, puntos y el símbolo @.

**Contraseña**    
Muy débil  
Importante: Necesitas esta contraseña para acceder. Por favor, guárdala en un lugar seguro.

**Confirma la contraseña**  Confirma el uso de una contraseña débil.

**Tu correo electrónico**   
Comprueba bien tu dirección de correo electrónico antes de continuar.

**Visibilidad en los motores de búsqueda**  Disuade a los motores de búsqueda de indexar este sitio.  
Depende de los motores de búsqueda atender esta petición o no.

Si lo hemos realizado todo correctamente, ya tendremos instalado nuestro wordpress de manera local.

Si queremos acceder, la manera con la que yo lo hago es introduciendo este enlace <http://localhost/wordpress/wordpress/wp-admin/index.php>

Este enlace nos lleva al menú principal donde introduciremos el usuario y la contraseña.



Si lo hemos hecho bien conseguiremos llegar a una página como esta:

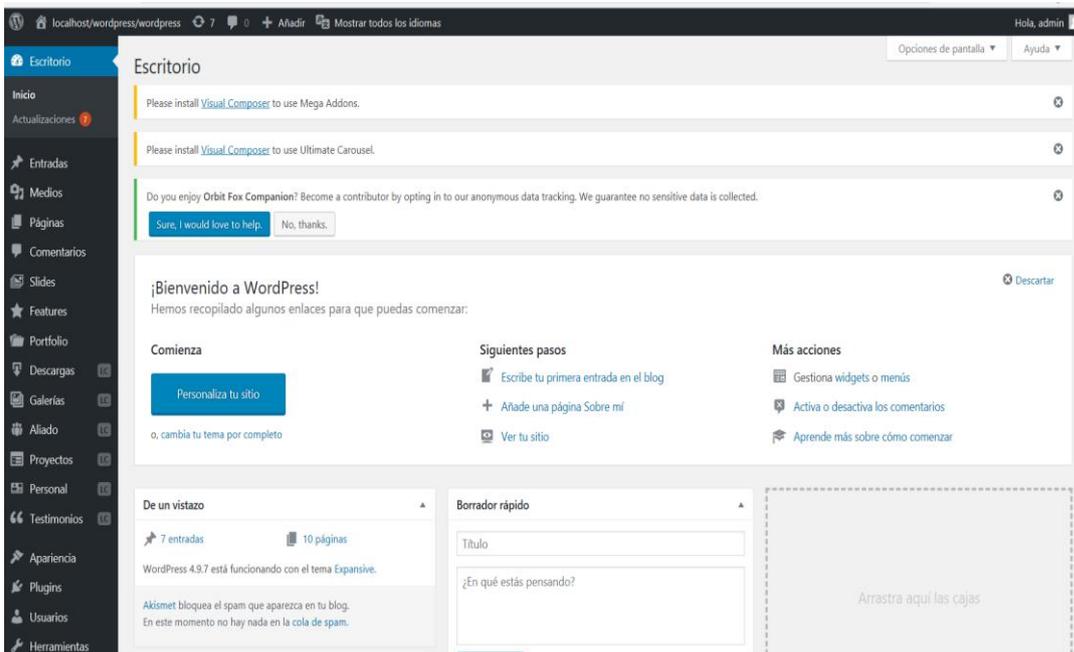


Imagen 15:Menú de Wordpress

Tabla comparativa de los temas:

Características	Expansive	Cenote	Metro CreativeX
Gratuito			
Pago			
Soporte			
Posibilidad de trabajar con <i>live composer</i>			
Editor integrado CCS			

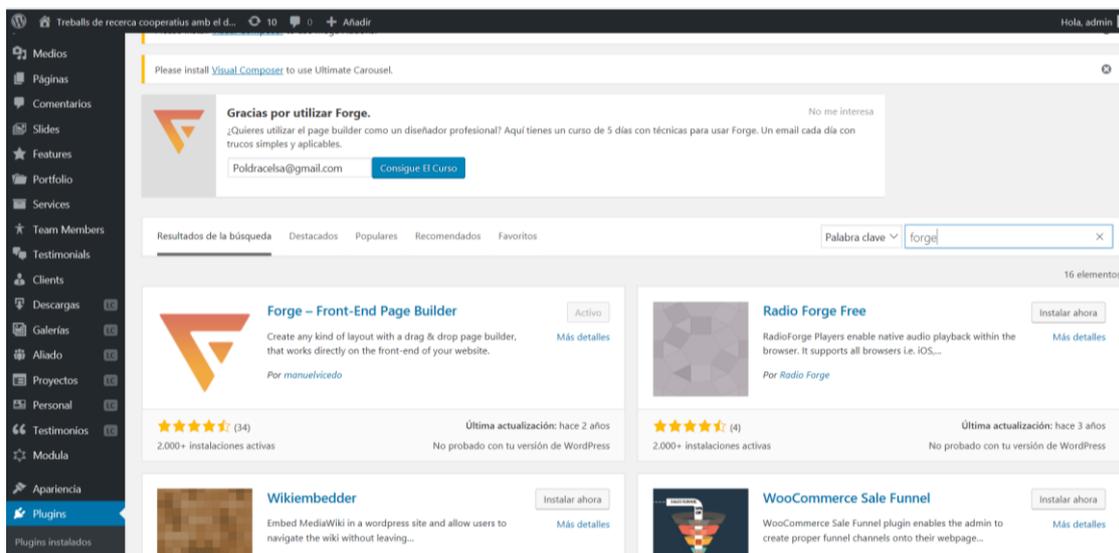
Finalmente para la página web se utilizó el tema Allegiant, ya que era el que se proponía en el tutorial que estuve siguiendo para la creación de la página.

## 14. Creación de una página Web

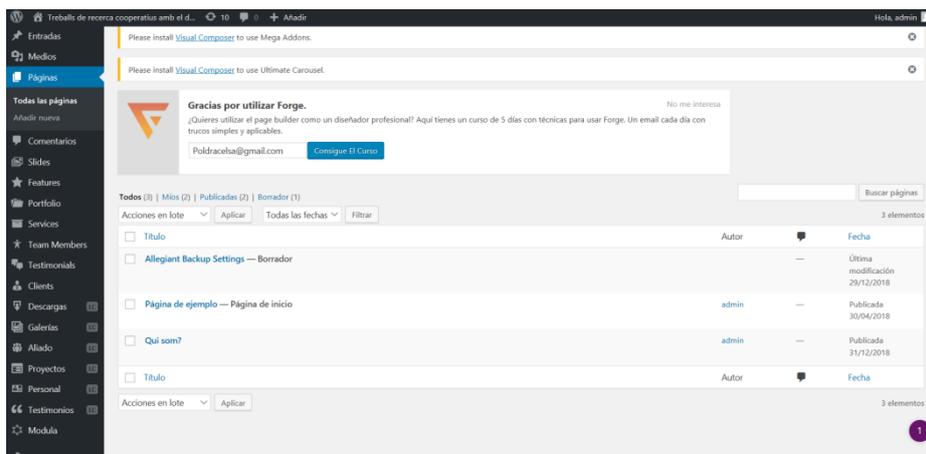
Para estructurar la página web lo que hice fue utilizar un “page builder” que te permite organizar la página sin necesidad de programar. Forge es un plugin gratuito que se encuentra en el repositorio oficial de Wordpress.

Este plugin lo usé combinado con el tema de Wordpress Allegiant que me permitía hacer varias modificaciones.

Primero instalé Forge y lo activé desde el administrador de pluggins de Wordpress.



Después creé una nueva página llamada “página de ejemplo”(la página que viene por defecto cuando instalas wordpress).



Esta será la página principal de la página web.

Deberemos eliminar la barra lateral de manera que quede mucho más espacio para el contenido de la página.

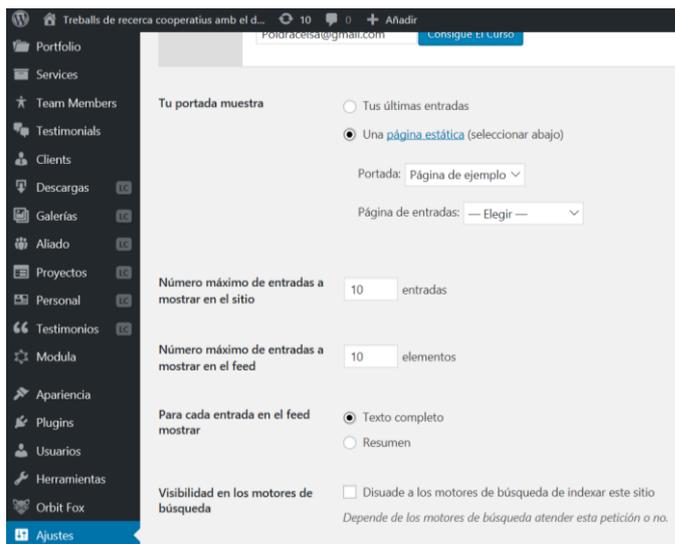
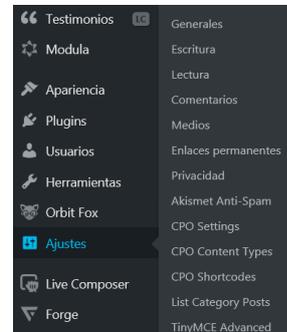
Posición de la Barra Lateral



Determina la ubicación de la barra lateral por defecto.

Ya habiendo modificado la estructura de la página debemos dirigirnos a ajustes y dirigirnos al apartado de lectura.

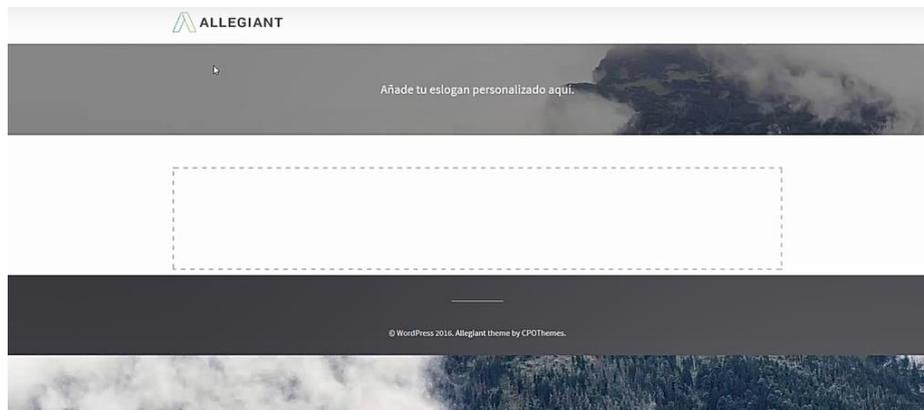
Allí podremos modificar nuestra portada y de este modo hacer que la portada muestre una página estática que en nuestro caso será la página de ejemplo y no las últimas publicaciones que hagamos.



Para usar Forge solamente tenemos que dirigirnos a nuestra página web.



Ya en la página web lo que debemos hacer es irnos a la barra de administración y allí veremos un icono llamado Forge page builder, lo clicaremos y se iniciará la interfaz de administración de Forge.

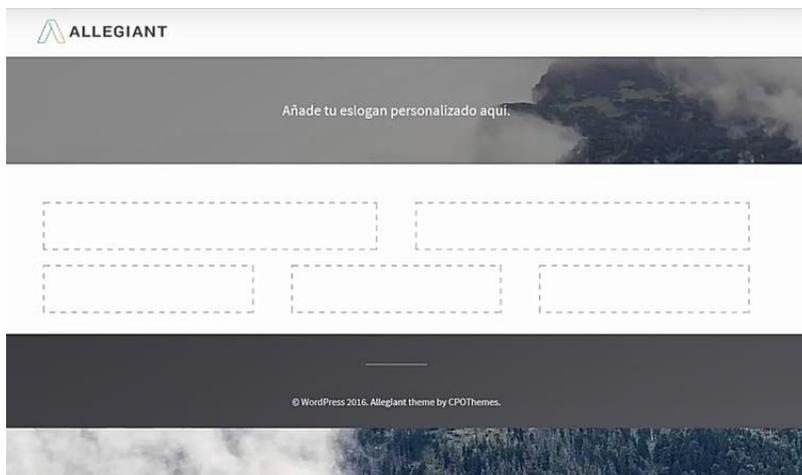


Inicialmente la página aparece vacía.

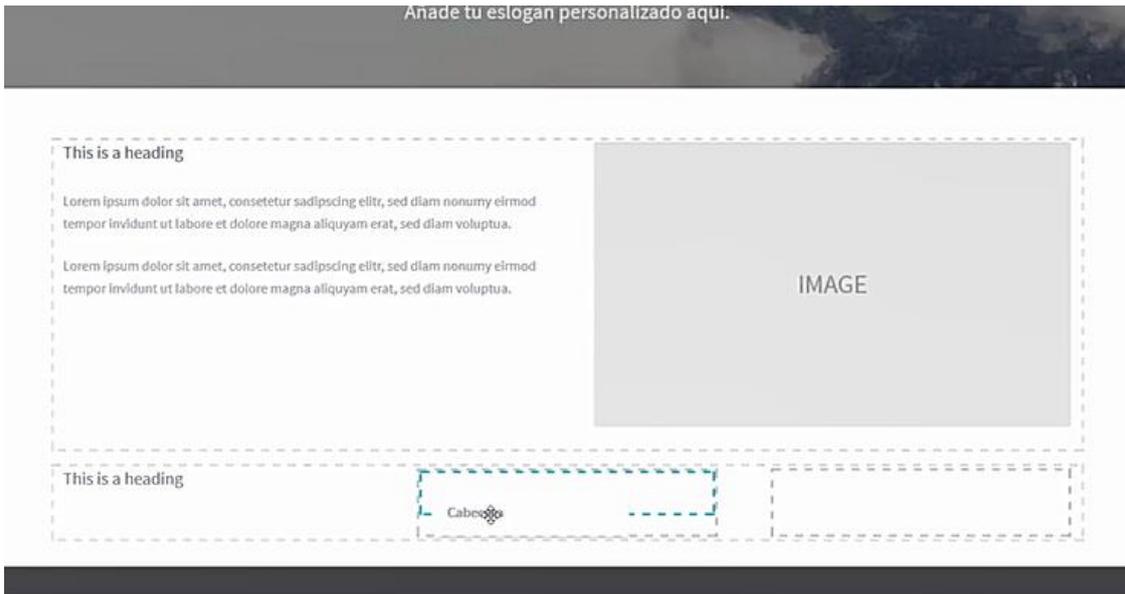
Con los diferentes elementos disponibles en el page builder podemos darle la estructura deseada. Yo he usado principalmente 4 elementos: Fila, Bloque de Texto, Cabecera e Imagen aunque hay muchos más disponibles.



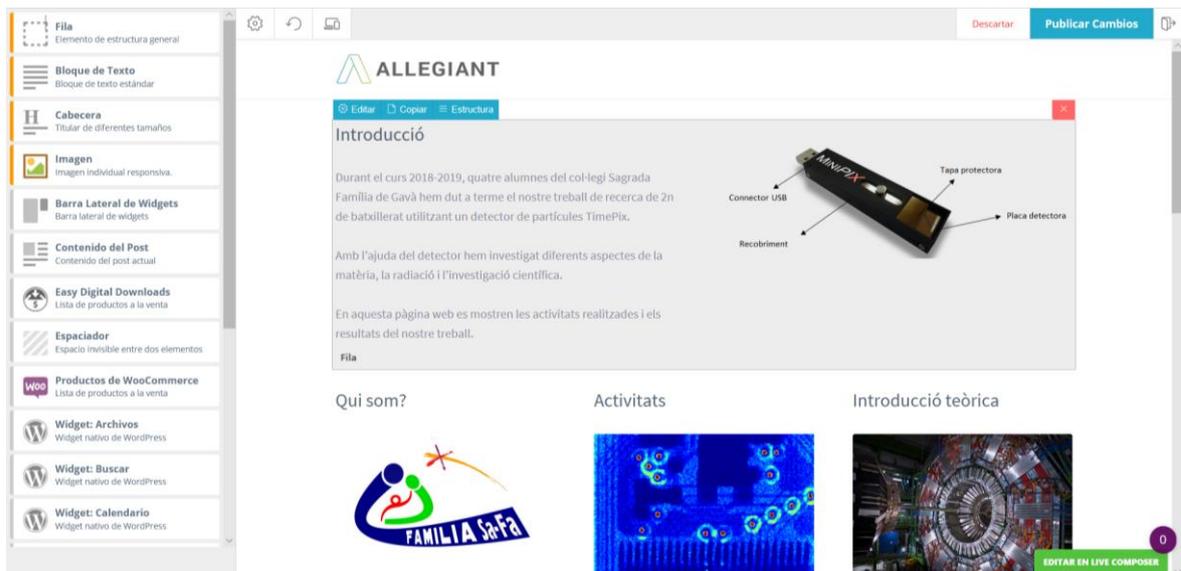
Con la herramienta Fila modificamos la cantidad de filas y columnas que aparecen en la página.



Con las herramientas Cabecera, Cuadro de Texto e Imagen modificamos cada uno de los bloques generados.



Editamos y modificamos los contenidos para añadir los contenidos de mi página.



Después de modificarla la página principal del sitio web se ve de la siguiente manera:

Treballs de recerca cooperatius amb el d... Personalitzar 10 Afiadir Editar página Forge Page Builder Hola, admin

### Introducció

Durant el curs 2018-2019, quatre alumnes del col·legi Sagrada Família de Gavà hem dat a terme el nostre treball de recerca de 2n de batxillerat utilitzant un detector de partícules TimePix.

Amb l'ajuda del detector hem investigat diferents aspectes de la matèria, la radiació i l'investigació científica.

En aquesta pàgina web es mostren les activitats realitzades i els resultats del nostre treball.

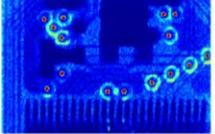
### Qui som?



Som el Mario Agustí, el Carles Vallés, el Guillem Galve i el Pol Marcos, alumnes de 2n de Batxillerat del col·legi Sagrada Família de Gavà.

En aquesta secció si fas clic a l'imatge ens podràs conèixer una mica més.

### Activitats



En aquests treballs de recerca s'han fet conjuntament diverses activitats complementàries, com per exemple, visitar el laboratori de física moderna de la UB per a fer uns experiments, presentar aquests experiments a la Barcelona TechnoWeek, o presentar algunes de les imatges preses amb el detector al concurs de fotografies de física de la Societat Catalana de Física.

### Introducció teòrica



Cada un dels alumnes ha fet una part teòrica referida al detector i la física que hi ha al darrera centrant-se en un aspecte concret.

Així s'han investigat els principis de física moderna i nuclear necessaris per a comprendre la radiació. Igualment s'ha analitzat el software i el hardware del

[EDITAR EN LIVE COMPPOSER](#)

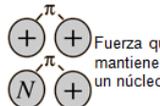
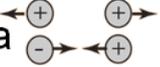
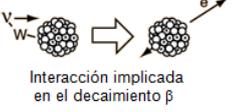
Desde la página principal se puede navegar a las diferentes páginas del sitio.

# 15. Parte teórica común

## Introducción a la física nuclear

### Fuerzas fundamentales:

Las fuerzas fundamentales son las cuatro fuerzas de interacción fundamentales que se encuentran en el universo y son la interacción gravitatoria, la electromagnética, la nuclear fuerte y por último la nuclear débil.

		Intensidad	Alcance (m)	Partícula
Fuerte		1	$10^{-15}$ (Diámetro del tamaño promedio del núcleo)	gluón $\pi$ (nucleones)
Electromagnética		$\frac{1}{137}$	Infinito	fotón masa = 0 espín = 1
Débil		$10^{-6}$	$10^{-18}$ (0,1% del diámetro del protón)	bosones vectoriales intermedios $W^+$ , $W^-$ , $Z_0$ masa > 80 GeV espín = 1
Gravitatoria		$6 \times 10^{-39}$	Infinito	gravitón $\zeta$ ? masa = 0 espín = 2

### Fuerzas fundamentales

### La interacción gravitatoria:

La gravedad es descrita con mayor precisión por la teoría general de la relatividad (propuesta por Albert Einstein en 1915) que describe la gravedad no como una fuerza, sino como una consecuencia de la curvatura del espacio-tiempo causada por la distribución desigual de la masa. El ejemplo más extremo de esta curvatura del espacio-tiempo es un agujero negro, del cual nada, ni siquiera la luz, puede escapar una vez que pasa el horizonte de eventos del agujero negro. Sin embargo, para la mayoría de las aplicaciones, la gravedad se aproxima bien a la ley de Newton de la gravitación universal, que describe la gravedad como una fuerza que hace que dos

cuerpos se atraigan entre sí, con la fuerza proporcional al producto de sus masas e inversamente proporcional al cuadrado de la distancia entre ellas.

La gravedad es la más débil de las cuatro fuerzas fundamentales de la física. Como consecuencia, no tiene influencia significativa a nivel de partículas subatómicas. En contraste, es la fuerza dominante en la escala macroscópica, y es la causa de la formación, forma y trayectoria de los cuerpos astronómicos. Por ejemplo, la gravedad hace que la Tierra y los otros planetas orbiten alrededor del Sol, también la Luna orbita la Tierra y provoca la formación de mareas, la formación y evolución del Sistema Solar, las estrellas y las galaxias.

### **Explicación del tirón gravitatorio según los dos modelos:**

Dos objetos ejercen una fuerza de atracción una sobre la otra conocida como "gravedad". Sir. Isaac Newton cuantificó la gravedad entre dos objetos cuando formuló sus tres leyes del movimiento. La fuerza que tira entre dos cuerpos depende de cuán masiva sea cada uno y de cuán alejados estén los dos. A pesar de que el centro de la Tierra te empuja hacia ella, tu centro de masa retrocede hacia la Tierra. Pero el cuerpo más masivo apenas siente el tirón de ti, mientras que con tu masa mucho más pequeña te encuentras firmemente arraigado gracias a esa misma fuerza. Sin embargo, las leyes de Newton suponen que la gravedad es una fuerza innata de un objeto que puede actuar sobre una distancia.

Albert Einstein, en su teoría de la relatividad especial, determinó que las leyes de la física son las mismas para todos los observadores que no aceleran, y demostró que la velocidad de la luz dentro de un vacío es la misma, sin importar la velocidad a la que viaje un observador. Como resultado, descubrió que el espacio y el tiempo estaban entrelazados en un solo continuo conocido como espacio-tiempo. Los eventos que ocurren al mismo tiempo para un observador podrían ocurrir en diferentes momentos para otro.

Mientras elaboraba las ecuaciones de su teoría general de la relatividad, Einstein se dio cuenta de que los objetos masivos causaban una distorsión en el espacio-tiempo. Para concebir este concepto se hace similar a colocar un gran cuerpo en el centro de una membrana. El cuerpo presionaría hacia abajo en la tela, causando que se forme un

hoyuelo. Una canica rodada alrededor del borde giraría en espiral hacia el interior del cuerpo, tirada de la misma manera que la gravedad de un planeta atrae a las rocas en el espacio.

### Interacció electromagnètica:

Para entender esta interacción es necesario saber que en la teoría atómico existen tres tipos de partículas que forman los átomos: los electrones, neutrones y protones. Estas partículas son caracterizadas por la masa y una carga eléctrica. Al igual que en la gravitatoria de un cuerpo con presencia de masa produce esta interacción en el electromagnetismo se produce gracias a estas cargas eléctricas.

En el electromagnetismo se diferencian dos tipos de fuerzas: una es la fuerza de repulsión es producida cuando dos cuerpos interaccionan entre sí con la misma carga y el otro es la de atracción que se da en caso de que dos cuerpos de diferente carga interaccionan entre sí.

Al final la fuerza electromagnética explica cómo interactúan las partículas cargadas en movimiento y estacionarias. Su nombre fuerza electromagnética es debido a que incluye la fuerza eléctrica y la fuerza magnética; De forma que las fuerzas magnéticas y las fuerzas eléctricas son en realidad la misma fuerza fundamental.

La fuerza eléctrica actúa entre todas las partículas cargadas, estén o no moviéndose. Y a diferencia la fuerza magnética actúa entre las partículas cargadas en movimiento. Esto significa que cada partícula cargada emite un campo eléctrico, se mueva o no. Las partículas cargadas en movimiento (como las de la corriente eléctrica ) emiten campos magnéticos .Einstein desarrolló su teoría de la relatividad a partir de la idea de que si el observador se mueve con las partículas cargadas, los campos magnéticos se transforman en campos eléctricos y viceversa.

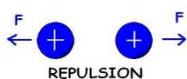


Figura 1

Repulsió i atracció entre càrregues elèctriques.

**Interacción nuclear fuerte:**

Esta fuerza a diferencia de las otras dos ya mencionadas se encuentra únicamente en situaciones de dimensiones atómicas muy pequeñas. Concretamente esta interacción es la responsable de mantener unidos los nucleones, es decir, los neutrones y protones dispuestos en el núcleo atómico venciendo así a las fuerzas de repulsión electromagnética producidas por la presencia de la misma carga positiva entre protones produciendo la unión entre neutrones y protones en el núcleo. Sobre esta fuerza hay que recalcar que sólo se encuentra a distancias muy pequeñas de la medida de núcleos atómicos. Bajo el Modelo Estándar, una de las partículas más pequeñas y fundamentales, es decir, una que no se puede dividir en partes más pequeñas, es el quark. Estas partículas son los bloques de construcción de una clase de partículas masivas conocidas como hadrones, que incluyen a los ya nombrados protones y neutrones. De forma que la interacción fuerte fundamental mantiene unidos los quarks constituyentes de un hadrón, y la fuerza residual mantiene juntos los hadrones, como el protón y los neutrones en un núcleo.

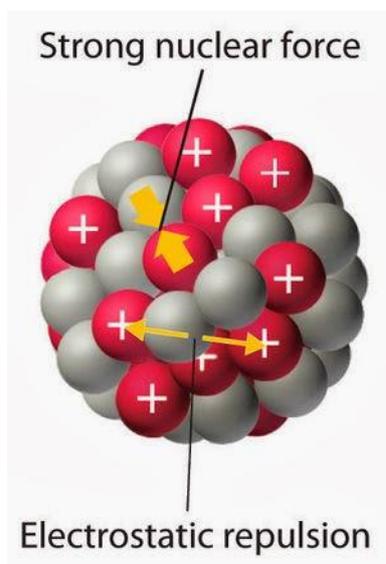
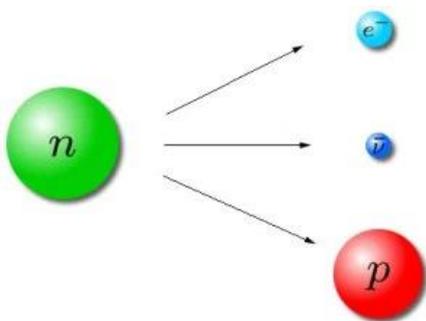


Diagrama del nucli d'un àtom amb les indicacions de les forces que estan presents (Repulsió electroestàtica i força nuclear forta)

**Interacción nuclear débil:**

Esta interacción define principalmente el concepto de las desintegraciones radiactivas. Como interacción débil no sólo puede ocasionar efectos puramente atractivos o repulsivos, sino que también puede producir el cambio de identidad de las partículas involucradas, es decir, lo que se conoce como una reacción de partículas subatómicas. En caso de la interacción débil, fermiones pueden intercambiar tres tipos distintos de portadores de fuerza conocidos como los bosones  $W^+$ ,  $W^-$  y  $Z$ . La masa de cada uno de estos bosones es mucho mayor que la masa de un protón o neutrón, lo cual es consistente con el corto alcance de la fuerza débil. De hecho, la fuerza se denomina débil porque su intensidad de campo en una distancia dada suele ser de varios órdenes de magnitud menor que la de la fuerza nuclear fuerte o la fuerza electromagnética.

Los quarks, que forman partículas compuestas como neutrones y protones, vienen en seis "sabores" (arriba, abajo, extraño, encanto, arriba y abajo) que dan a esas partículas compuestas sus propiedades. La interacción débil es única porque permite que los quarks intercambien su sabor por otro. El intercambio de esas propiedades está mediado por los bosones portadores de fuerza. Por ejemplo, durante la desintegración beta menos, un quark down dentro de un neutrón se transforma en un quark up, convirtiendo así el neutrón en un protón y dando como resultado la emisión de un electrón y un antineutrino electrónico.



Desintegració del tipus beta d'un neutró

### **Radioactivitat a les partícules:**

El concepto de radiación es la propagación de energía en forma de ondas electromagnéticas o partículas subatómicas a través del vacío o de un medio material. Y esta se puede subdividir en dos principales tipos, la radiación no ionizante y la radiación ionizante.

### **Radiación no ionizante:**

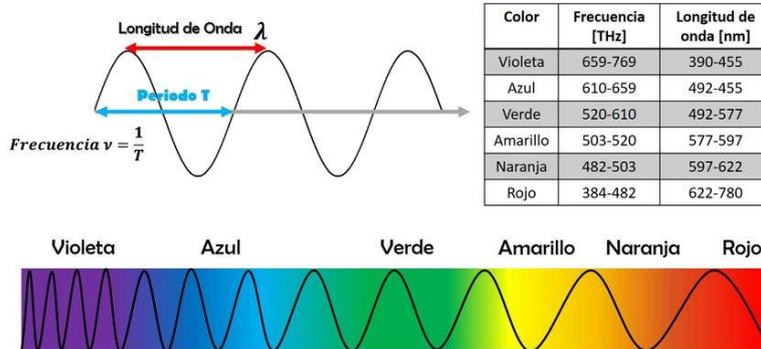
En cuanto a su definición, se refiere a cualquier tipo de radiación electromagnética que no transporta suficientes cuantos de energía para ionizar átomos o moléculas, es decir, para eliminar completamente un electrón de un átomo o molécula. En lugar de producir iones cargados al pasar a través de la materia como la radiación ionizante, la radiación electromagnética no ionizante tiene suficiente energía solo para la excitación, el movimiento de un electrón hacia un estado de energía superior. En cuanto a tipos existe una amplia variedad de radiaciones no ionizantes, pero las principales y con las que más convivimos en nuestro día a día son:

### **Ondas de radió:**

Las ondas de radio son un tipo de radiación electromagnética con longitudes de onda en el espectro electromagnético más largas que la luz infrarroja. Como todas las demás ondas electromagnéticas, viajan a la velocidad de la luz. Las ondas de radio que ocurren naturalmente son hechas por rayos o por objetos astronómicos. En cuanto a sus usos en la tecnología las ondas de radio generadas artificialmente se utilizan para comunicaciones de radio fijas y móviles, radiodifusión, radar y otros sistemas de navegación, comunicación por satélite, redes de computadoras e innumerables otras aplicaciones. Además estas según las diferentes frecuencias de las ondas de radio tienen diferentes características y propiedades de propagación en la atmósfera de la Tierra; por ejemplo, las ondas largas pueden cubrir una parte de la Tierra de manera muy consistente, las ondas más cortas pueden reflejarse en la ionosfera y viajar alrededor del mundo, y las longitudes de onda mucho más cortas se doblan o reflejan muy poco y viajan en una línea de visión.

### **Luz del espectro visible (Luz visible):**

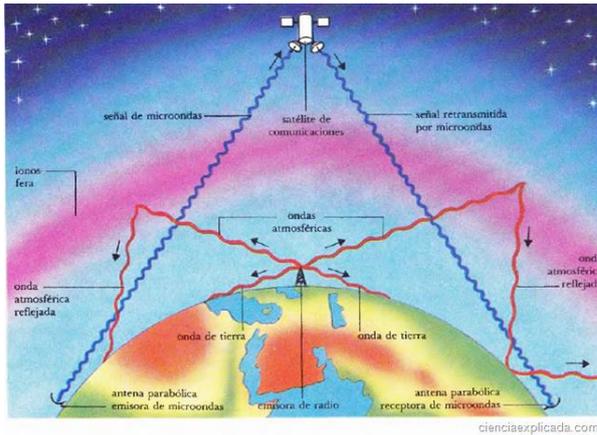
La luz, o luz visible, es el rango muy estrecho de la radiación electromagnética que es visible para el ojo humano con una longitud de onda de aproximadamente 400–700 nm, o hasta 380–750 nm.



**Microondas:**

Las microondas son una forma de radiación electromagnética con longitudes de onda que van desde aproximadamente un metro a un milímetro

Las microondas viajan únicamente por caminos de línea de visión a diferencia de las ondas de radio de baja frecuencia. Además no viajan como ondas terrestres que siguen el contorno de la Tierra, ni se reflejan en la ionosfera. En cuanto a sus usos la radio de microondas se utiliza en transmisiones de radiodifusión y telecomunicaciones, además tienen un gran protagonismo en el ámbito de la astronomía radial debido a la recepción de microondas emitidas por cuerpos celestes lejanos como planetas, estrellas, galaxias, etc... también se encuentra el uso del microondas en el calentamiento dieléctrico principalmente por la absorción de la energía en el agua que contengan los cuerpos sometidos a las microondas.



### Radiación ionizante:

Las radiaciones ionizantes son aquellas formadas por fotones o partículas que interactúan con la materia y producen iones, como ejemplo de radiaciones ionizantes por partículas o corpusculares son ejemplos la radiactividad alfa y la radiactividad beta ya mencionadas anteriormente y respecto a radiaciones electromagnéticas también son ejemplos desde los rayos ultravioleta a los rayos gamma o los rayos X.

Estas radiaciones en cuanto a su procedencia hay varias, principalmente se encuentran en sustancias radiactivas que emiten estos tipo de radiaciones de forma espontánea o también se pueden encontrar estas radiaciones en generadores artificiales como podría ser una máquina de rayos X o en un acelerador de partículas.

La utilización de estas es muy importante hoy en día para el desarrollo de la industria y medicina. En industria se han podido encontrar formas de conseguir grandes cantidades de energía gracias a la fisión nuclear, o en el caso de la medicina se encuentran diversos procedimientos mediante radiaciones ionizantes para diagnosticar enfermedades o tratamientos como puede ser la radioterapia.

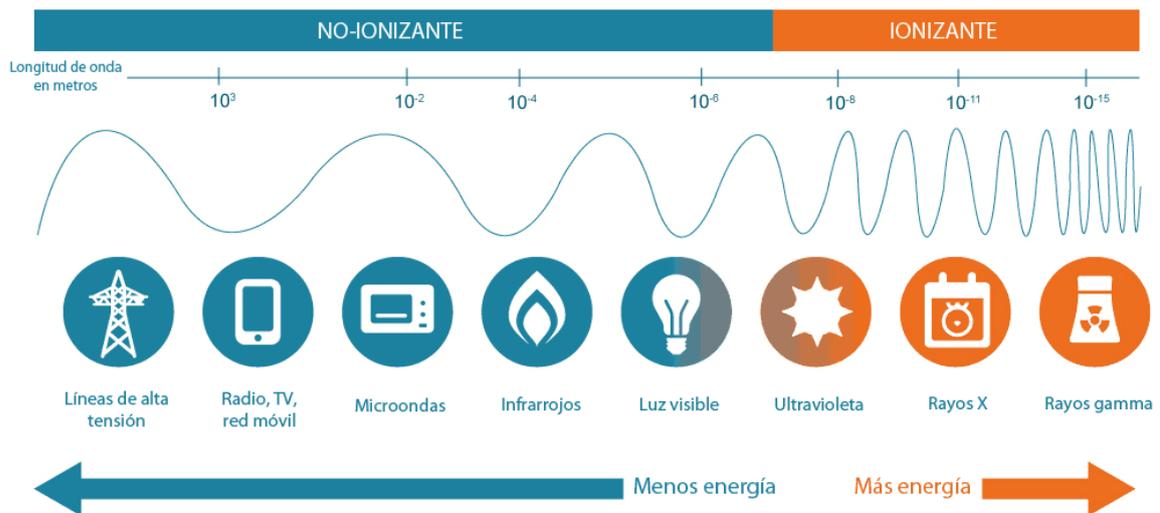


Diagrama de radiaciones según su energía y longitud de onda.

La radiación ionizante se clasifica por la naturaleza de las partículas u ondas electromagnéticas que crean el efecto ionizante. Estos tienen diferentes mecanismos de ionización, y pueden agruparse como ionizantes directos o indirectamente.

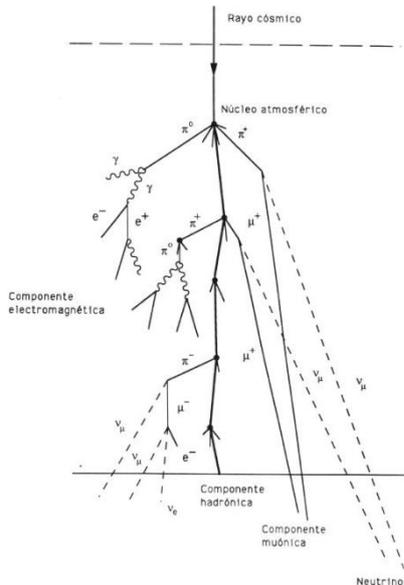
**Radiación directamente ionizante:**

Cualquier partícula masiva cargada puede ionizar átomos directamente por interacción fundamental a través de la fuerza de Coulomb si transporta suficiente energía cinética. Esto incluye núcleos atómicos, electrones, muones, piones cargados, protones y núcleos cargados de energía despojados de sus electrones.

Cuando se mueven a velocidades relativistas, estas partículas tienen suficiente energía cinética para ser ionizantes, pero no se requieren velocidades relativistas. Por ejemplo, una partícula alfa típica es ionizante, pero se mueve a aproximadamente 5% c, y un electrón con 33 eV (suficiente para ionizar) se mueve a aproximadamente 1% c.

Los rayos cósmicos naturales se componen principalmente de protones relativistas, pero también incluyen núcleos atómicos más pesados como los iones de helio y los iones HZE. En la atmósfera, tales partículas a menudo son detenidas por moléculas de aire, y esto produce piones cargados de corta duración, que pronto se descomponen

en muones, un tipo primario de radiación de rayos cósmicos que llega al suelo. Además, los piones también se pueden producir en grandes cantidades en aceleradores de partículas.



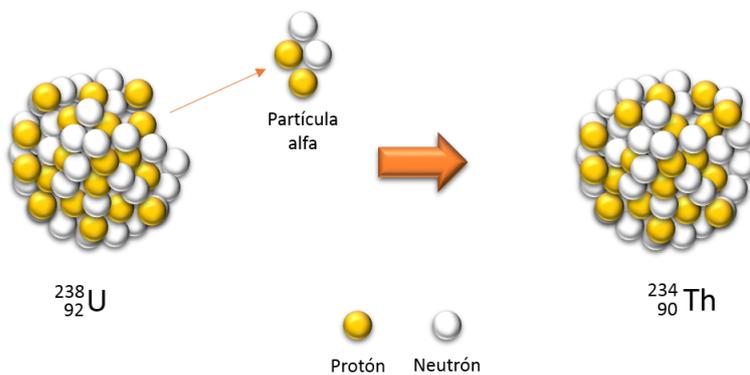
### La radiación alfa:

La radiación alfa está constituida por núcleos de helio ionizados ( $\text{He}^{2+}$ ), ya que, no presentan electrones ( $e^-$ ) lanzados a una gran velocidad aproximadamente  $1/20c$  (on “c” es la velocidad de la luz).

Son una forma altamente ionizante de la radiación de partículas, aunque tienen una profundidad de penetración baja, ya que puede ser detenida por unos pocos centímetros de aire, o por la piel.

Para la organización de radiación alfa se debe dar la desintegración alfa la cual es un tipo de desintegración radiactiva en la que un núcleo atómico emite una partícula alfa (núcleo de helio) y, por lo tanto, se transforma o 'descompone' en un núcleo atómico diferente, con un número de masa que se reduce en cuatro y un número atómico que

se reduce en dos. La energía de las partículas alfa emitidas varía, con las partículas alfa de mayor energía emitidas desde núcleos más grandes, pero la mayoría de las partículas alfa tienen energías de entre 3 y 7 MeV (mega-electron-voltios), correspondientes a vidas medias extremadamente largas y extremadamente cortas. Esta energía es una cantidad sustancial de energía para una sola partícula, pero su gran masa significa que las partículas alfa tienen una velocidad más baja (con una energía cinética típica de 5 MeV; la velocidad es de aproximadamente 15,000 km / s, que es el 5% de la velocidad de luz) que cualquier otro tipo común de radiación (partículas  $\beta$ , neutrones, etc.) Debido a su carga y gran masa, las partículas alfa son absorbidas fácilmente por los materiales y pueden viajar solo unos pocos centímetros en el aire.



Proceso de desintegración alfa en un núcleo de Uranio 238(U238) que da lugar a una partícula alfa, y el cambio a un núcleo de Torio 234(Th234)

### La radiación beta:

La radiación beta está constituida por partículas beta, que en este caso no son más que electrones a velocidades muy cercanas a las de la luz, aproximadamente  $0,99c$  (donde "c" es la velocidad de la luz).

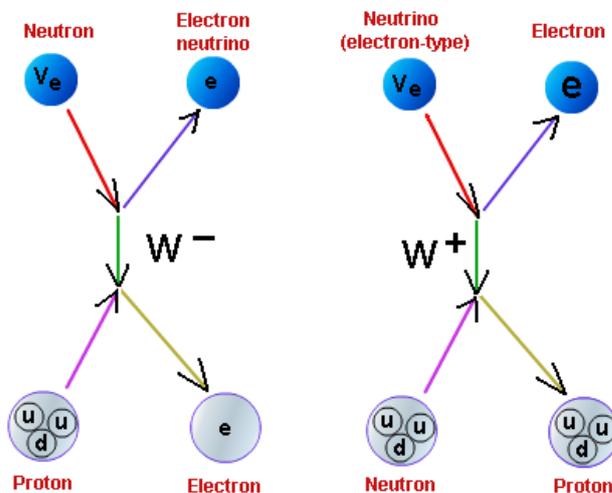
Más adelante se descubrió que aparte de la emisión de radiación beta para electrones a altas velocidades también existía de una forma de reacción inducida artificialmente la aparición de una radiación beta compuesta por positrones (antipartícula del electrón) a la misma velocidad que el electrón.

El proceso en el que un neutrón se transforma en protón y viceversa se denomina desintegración beta. La desintegración beta se produce cuando, en un núcleo con

demasiados protones o demasiados neutrones, de forma que uno de los protones o neutrones se transforma en el otro, y para la emisión de estas partículas se ha de producir una reacción nuclear la cual puede ser de dos formas:

De los tres tipos comunes de radiación emitida por los materiales radiactivos, la beta tiene un poder de penetración medio y el poder de ionización dentro de sus competentes también catalogado como medio. Aunque las partículas beta emitidas por diferentes materiales radiactivos varían en energía, la mayoría de las partículas beta pueden detenerse unos milímetros de aluminio. Sin embargo, esto no significa que los isótopos emisores beta puedan ser completamente protegidos por tales escudos delgados: a medida que se desaceleran en la materia, los electrones beta emiten rayos gamma secundarios, que son más penetrantes que los betas de por sí.

Al estar compuesto de partículas cargadas, la radiación beta es más fuertemente ionizante que la radiación gamma. Y es así que esta cuando pasa a través de la materia, una partícula beta es desacelerada por las interacciones electromagnéticas y puede emitir rayos X bremsstrahlung .



$\beta^+$ , un protón se descompone en un neutrón, un positrón y un neutrino.

$\beta^-$ , descomposición de un neutrón que se desintegra en un protón, un electrón y un antineutrino.

### Representació dels dos tipus de radiacions beta (+/-) En un procés d'anihilació

### **Núcleos cargados:**

Los núcleos cargados son característicos de los rayos cósmicos galácticos y los eventos de partículas solares y, a excepción de las partículas alfa, no tienen fuentes naturales en la Tierra. Sin embargo, en el espacio, los protones de muy alta energía, los núcleos de helio y los iones HZE pueden detenerse inicialmente mediante capas relativamente delgadas de protección, ropa o piel. Por otra parte, la interacción resultante generará radiación secundaria. En caso de que colisionara con nuestro tejido biológico, si solo un átomo de tejido es desplazado por un protón energético, la colisión causará más interacciones en el cuerpo. Y a este suceso se le denomina transferencia de energía lineal (LET).

Para tener una idea de este suceso LET se puede visualizar como una bola de billar que golpea a otra conservando el impulso inicial, enviando así a ambas con la energía de la primera bola dividida entre las dos de manera desigual. Pues lo mismo pasa cuando un núcleo cargado golpea un núcleo de movimiento relativamente lento de un objeto en el espacio, se produce LET y las colisiones liberan neutrones, partículas alfa, protones de baja energía y otros núcleos que contribuyen a la dosis total absorbida por el tejido.

### ***Positrones y otros tipos de antimateria:***

El positrón o antielectrón es la antipartícula o la contraparte de antimateria del electrón. Cuando un positrón de baja energía choca con un electrón de baja energía, se produce la aniquilación, lo que resulta en su conversión en la energía de dos o más fotones de rayos gamma.

También los positrones pueden generarse por desintegración nuclear por emisión de positrones (a través de interacciones débiles) o por producción de pares a partir de un fotón suficientemente energético. En cuanto a una aplicación de los positrones son las fuentes artificiales comunes de radiación ionizante que se utilizan en las tomografías médicas por tomografía por emisión de positrones (TEP).

Además, como los positrones son partículas cargadas positivamente, también pueden ionizar directamente un átomo a través de las interacciones de Coulomb.

### Radiación indirectamente ionizante:

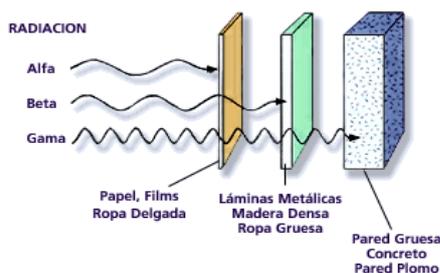
La radiación ionizante indirecta es eléctricamente neutra y, por lo tanto, no interactúa fuertemente con la materia. La mayor parte de los efectos de la ionización se deben a ionizaciones secundarias.

### Radiación gamma:

La radiación gamma es un tipo de radiación electromagnética, que por tanto, es capaz de propagarse por el vacío a la velocidad de la luz ( $c$ ). Y normalmente suelen tener una energía equivalente a un megaelectronvoltio (MeV). La característica de esta radiación es que suele estar ligada a la radiación alfa o beta debido a que cuando se produce una de estas para la estabilidad del núcleo el diferencial de energía que se pierde se expulsa en forma de ondas electromagnéticas, en este caso por radiación gamma.

Debido a las altas energías que poseen, los rayos gamma constituyen un tipo de radiación ionizante capaz de penetrar en la materia más profundamente que la radiación alfa y la beta. Es así, que pueden llegar a causar grave daño al núcleo de las células, por lo cual esta radiación se suele usar para esterilizar equipos médicos y alimentos. Los rayos gamma se producen durante la descomposición gamma, que normalmente ocurre después de que ocurren otras formas de descomposición, como la desintegración alfa o beta.

La descomposición gamma también puede seguir reacciones nucleares como la captura de neutrones, la fisión nuclear o la fusión nuclear. La desintegración gamma es también un modo de relajación de muchos estados excitados de núcleos atómicos que siguen a otros tipos de desintegración radiactiva, como la desintegración beta, siempre que estos estados posean el componente necesario del espín nuclear.



## Els Fonaments de la Física Moderna

### -Models de la llum:

#### Discrepància entre models: Model ondulatori i model corpuscular:

Durant tota la història, la llum va ser interpretada per dos models diferents. Un d'aquests era el model ondulatori, el qual defensava que aquesta es comportava com a un conjunt d'ones, en canvi, l'altre model, el model corpuscular, defensava l'existència d'una estructura interna de la llum més definida, conformada per diversos feixos de partícules, anomenades fotons.

En les següents taules, podem observar clarament les característiques de cadascun d'aquests models, a més a més dels fets que apunten al compliment de cadascun, i els fets que els fan abolir:

#### -Característiques dels models:

<b>MODEL CORPUSCULAR</b>	<b>MODEL ONDULATORI</b>
Proposat per Isaac Newton (1704)	Proposat per Cristian Huygens (1670)
La llum està formada per partícules, emeses en totes direccions i a gran velocitat	La llum està formada per un conjunt d'ones mecàniques, que es propagaven per mitjà d'un medi material de gran elasticitat.
Aquestes partícules produeixen la visió en arribar al nostre ull, estimulades per un moviment rectilini uniforme	Defensen l'existència de l'èter, el qual és definit com una substància imperceptible, present a tota zona de l'univers
Els corpuscles de diferent naturalesa es combinen per a formar els diferents colors	Les ones creen els diferents colors existents a partir de les seves respectives longituds d'ona, fenomen mesurat per Thomas Young

#### -Fets que confirmen cadascun dels models:

<b>FETS QUE CONFIRMEN EL MODEL CORPUSCULAR</b>	<b>FETS QUE CONFIRMEN EL MODEL ONDULATORI</b>
<u>Formació d'ombres:</u> Les ombres projectades sobre els objectes són nítides, fet per culpa	<u>Lleis de la refracció i la reflexió:</u> Aquestes poden ser demostrades fàcilment amb el

del rebot i a la continuació del moviment dels corpuscles que formen la llum, a l'instant d'impactar contra el cos.	principi de Huygens, que enuncia l'apropament del raig refractat a la normal quan el raig de llum passa a un medi amb major densitat.
<u>El camp gravitatori no sembla afectar la trajectòria rectilínia de la llum:</u> Això és degut a la gran velocitat de la llum, que fa que els corpuscles no es vegin afectats pels diferents camps de gravitació.	<u>Colors que formen la llum blanca:</u> Això, és fàcilment raonable si es considera que cadascun dels colors correspon a una longitud d'ona diferent; així, aquest fenomen seria conseqüència dels processos de difracció i refracció lumínica.
<u>Llei de la reflexió:</u> Es pot extrapolar com a una conseqüència d'un xoc elàstic, dut a terme pels corpuscles en impactar amb la superfície reflectora. A més, es poden aplicar els principis de conservació de la quantitat de moviment, juntament amb l'energia.	<u>Dos raigs de llum es creuen sense pertorbar la seva trajectòria:</u> Això és fàcilment demostrable si considerem la llum com un gran conjunt d'ones mecàniques, sense massa, pel que no desviaran les seves respectives trajectòries.

-Fets que deneguen la veracitat dels diferents models:

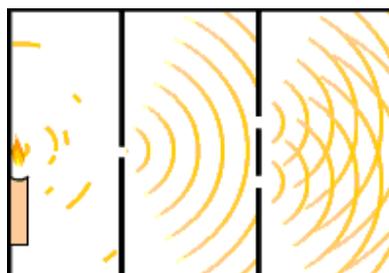
<b>FETS QUE REFUTEN EL MODEL CORPUSCULAR</b>	<b>FETS QUE REFUTEN EL MODEL ONDULATORI</b>
Si veritablement la llum fos un gran feix de partícules, aquesta hauria de perdre massa de forma considerable, aquest fet no és perceptible a la realitat.	La llum no presenta la capacitat de crear obstacles, pels quals desviarà la seva trajectòria, per mitjà de la difracció. Aquest fenomen és propi d'ones.
Si aquest model fos vàlid completament, els raigs de llum secants entre si haurien de desviar les seves respectives trajectòries, fet que no és donat a terme a la realitat.	La propagació de la llum en forma d'ones a través dels medis materials pot ser explicada com la vibració de les respectives partícules del medi al pas del raig de llum. En canvi, aquesta explicació no és acceptable pels raigs de llum provinents del Sol.
Segons aquest model, els processos de reflexió i refracció en un mateix sistema d'estudi són inexplicables, ja que no existia una explicació	

<p>raonable per la qual alguns corpuscles rebotaven davant la superfície reflectora i altres seguien la seva trajectòria al segon medi.</p>	
<p>Quan Newton dedueix el procés de refracció a partir del model corpuscular, dedueix que el raig refractat s'allunya de la normal quan el raig passa d'un medi a un altre més dens, fet que a la realitat és totalment el contrari.</p>	

Al segle XIX, per mitjà d'un experiment realitzat pel físic Thomas Young, molt conegut pel nom de l'experiment de la doble escletxa, es va aconseguir establir que la llum tenia característiques d'un caràcter plenament ondulatori, refutant així, fins a principis del segle XX, el model corpuscular. Aquest experiment, realitzat l'any 1801, va consistir en el següent:

En una habitació fosca, Young va fer entrar un raig ben fi de llum solar, a partir d'un mirall exterior i un forat a la finestra de l'habitació; aquest raig, va ser dirigit a una fina làmina de cartró de perfil respecte al raig, en el qual es podien diferenciar dos tipus de franges: les franges clares i les franges fosques, reflectides a la paret oposada a la làmina, i a cadascun dels forats que els corresponia; aquests, formaven un patró d'interferència, constituït per punts d'interferència constructiva i destructiva.

Si analitzem els resultats de l'experiment, observem que la làmina va separar el raig de llum primari en dues parts o subraigs, aquests difracten en impactar amb aquesta, formant així el patró d'interferència. A partir d'aquest fet, Young va mesurar el gruix de la làmina, les distàncies entre franges i la distància de la làmina a la paret, i aplicant les condicions d'interferència constructiva i destructiva va ser capaç de calcular la longitud d'ona dels subraigs.



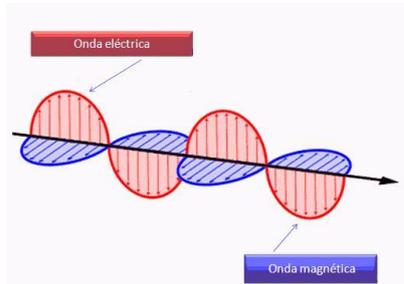
Imatge de l'experiment de la doble escletxa

**-Electromagnetisme:**

Inconvenients del model ondulatori:

La confirmació del model ondulatori va fer acceptar diferents fets especulats amb la veracitat d'aquest model. El principal d'aquests fets acceptats va ser l'existència de l'èter, com a substància present a tot l'univers; aquest presentava molts inconvenients, ja que aquesta havia de ser molt densa, per permetre la propagació de la llum com a ona transversal i a gran velocitat, i al mateix temps molt tènue, per no arribar a dificultar el moviment dels planetes i dels altres astres. Aquests inconvenients van ser resolts en gran part per James Clerk Maxwell, l'any 1873; aquest, a partir dels fonaments establerts per Faraday, Ampère i Thomson, va dur a terme un estudi sobre els fenòmens elèctrics i magnètics, que, en principi, no guardaven relació amb els fenòmens lluminosos. Els científics anteriorment esmentats, van imposar les bases dels camps elèctrics i els camps magnètics; Maxwell, en canvi, va suposar que aquests dos camps no eren independents entre sí, sinó que cadascun d'ells són originats sota determinades circumstàncies, sent originades per la mateixa magnitud física, la càrrega elèctrica.

A partir d'aquest estudi, Maxwell va poder establir les seves quatre lleis, i combinant-les entre sí, va obtenir una equació semblant a la d'una ona mecànica transversal, però amb un camp elèctric oscil·lant perpendicular a un camp magnètic també oscil·lant. Gràcies a aquest fet, va poder establir que els camps electromagnètics es propagaven per ones, a una velocitat  $c$ , equivalent a la velocitat de la llum.



Imatge que mostra l'estructura esquemàtica presentada per les ones electromagnètiques

Així, Maxwell va aconseguir unir les ones electromagnètiques amb els fets lluminosos. La llum, a partir de la publicació de les lleis de Maxwell, va ser considerada com a una ona electromagnètica, fet que semblava explicar la propagació d'aquesta, i que semblava reforçar la idea de l'existència de l'èter.

Llavors, amb les lleis de Maxwell es podia calcular la velocitat de la llum en un medi determinat ( $v$ ), a partir del coneixement de la permeabilitat elèctrica ( $\epsilon$ ) i magnètica ( $\mu$ ), responnent a la següent fórmula:

$$v = 1/\sqrt{\epsilon\mu}$$

Responent a aquesta mateixa fórmula, al buit, la velocitat de la llum s'expressa amb la lletra c:

$$c=2,9986 \cdot 10^8 \text{ m/s}$$

Aquest valor també és vàlid a l'aire, ja que posseeix les mateixes constants elèctriques del buit.

### **-Trencament de la visió mecanicista: El naixement de la física moderna:**

La física clàssica, en estar basada els principis del moviment establerts per Newton l'any 1687, estava dotada d'un caràcter altament mecànic, fet que va ser trencat definitivament amb la publicació de les lleis de Maxwell, ja que va suposar l'existència d'un nou fet físic, fet que no podia ser explicat per termes de matèria i moviment, aquest fet era la radiació.

Segons les lleis de Maxwell, dita radiació havia de ser emesa quan s'accelerava un sistema compost per partícules elèctriques, aquesta emissió junt amb l'absorció d'energia, segons les mateixes lleis, havia de produir-se de manera contínua, en canvi, més endavant, a partir de diferents mètodes experimentals, es va demostrar tot el contrari, es va demostrar l'existència de la discontinuïtat de l'emissió i absorció de la llum, de manera que recordava a una espècie de paquets. Paradoxalment, si considerem cadascun d'aquests paquets com a una partícula, observem que la llum, en ser absorbida o emesa per la matèria, ho fa com si estès constituïda per partícules, fet inexplicable per la teoria de Maxwell.

Aquest fet, més endavant seria conegut com a dualitat ona-còrpuscle, o dualitat ona-partícula, que juntament amb els avenços en l'estudi de la radioactivitat i amb l'aparició de la Teoria de la Relativitat, van donar lloc al gran avenç físic del segle XX, el qual va originar la física moderna.

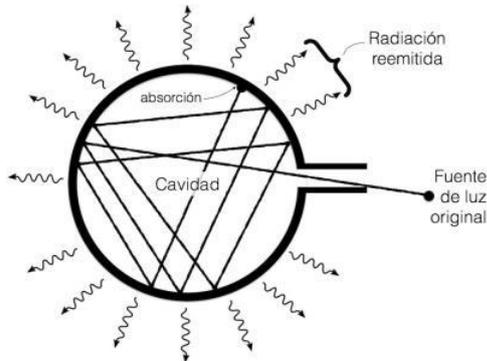
### **Radiació del cos negre:**

Per entendre aquest apartat, cal entendre que l'augment de la temperatura d'un cos fins a cert punt, provoca que l'espectre d'emissió del dit cos augmenti, pel que les ones emeses per aquest, posseiran una longitud d'ona més petita. A més, qualsevol cos absorbeix una certa part de la radiació imposada sobre aquest, i en reflecteix la part restant.

A principis del segle XX, les lleis de Maxwell havien creat un ingent interès en l'estudi de la radiació a tota la comunitat científica, pel que els científics van imposar, hipotèticament, l'existència d'un cos que absorbís tota la radiació que rebés, un cos ideal, el cos negre.

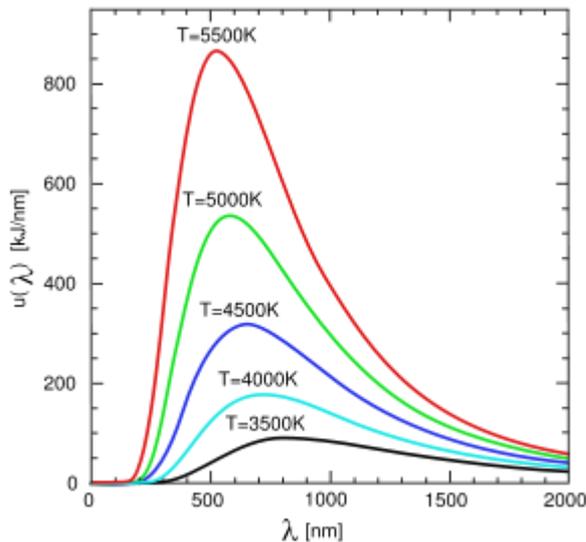
En la pràctica, dit cos és pràcticament existent, ja que, si fem una petita obertura a un cos que té les seves parets interiors recobertes de pols negra, dita obertura es comportarà com el cos anteriorment esmentat, ja que tota la radiació dirigida cap

aquesta, seria absorbida, com a conseqüència de la reflexió d'aquesta a les seves parets interiors. Aquest cos s'aproxima molt a la condició de cos negre, però no ho arriba a ser, ja que tot i que la probabilitat que la dita radiació sigui expulsada novament per l'obertura és ínfima, un cert nombre de radiació acabarà sent expulsada del cos.



Imatge esquemàtica del cos negre experimental

Si estudiem l'espectre de radiació emès per un cos negre per a cada temperatura, representant dit fet en una gràfica que relacioni les energies irradiades pel cos i les longituds d'ona de les dites radiacions, obtenim les corbes experimentals corresponents a la radiació del cos negre:



Imatge que mostra les corbes de radiació d'un cos negre

Per tant, la radiació havia d'estar formada per ones d'una longitud d'ona molt curta, com per exemple de radiació ultraviolada, aquest fet va ser conegut com a catàstrofe de l'ultraviolat.

L'any 1900, el físic alemany Max Planck, va deduir les corbes anteriorment esmentades teòricament. Per fer-ho, va suposar que l'energia radiada per un cos era absorbida i emesa d'una manera discontinua, en forma de paquets, que aquest va anomenar quants. L'energia de cadascun d'aquests quants responia a l'equació següent

$$E=hf$$

on  $h$  és la constant de Planck i  $f$  és la freqüència de radiació

Pel que l'energia completa de la radiació serà un múltiple de l'energia d'un únic quant.

La hipòtesi de Planck va fer retornar el model corpuscular de la llum, ja que cadascun dels quants que conformaven la llum podien ser considerats com a partícules, aquesta idea va ser recuperada uns anys més tard per Albert Einstein qui va utilitzar dita hipòtesi per demostrar teòricament l'efecte fotoelèctric.

### L'efecte fotoelèctric

L'experiment de l'efecte fotoelèctric consistia en un tub de vidre o quars, en el que es va recrear el buit, a l'interior d'aquest tub, hi havia dos elèctrodes, un càtode i un ànode. Quan aquest experiment es va completar, es va crear un corrent elèctric, el qual va ser creat per un voltatge  $V$ , molt gran, com per arrencar els electrons i dirigir-los cap a l'ànode. Aquest escapament és provocat pel transport d'energia d'una ona electromagnètica per part dels electrons del metall incidit, amb aquesta energia els electrons s'escapen dels àtoms dels metalls, passant a ser anomenats fotoelectrons.

Aquest fenomen era inexplicable per a la física clàssica, ja que, a partir d'un estudi detallat d'aquest, es van poder establir les següents conclusions

- Quan s'augmenta la intensitat de radiació augmenta la intensitat del corrent elèctric
- L'emissió fotoelectrònica és gairebé instantània, encara que la intensitat de radiació sigui molt petita
- L'emissió fotoelectrònica per a un metall determinat només té lloc si la freqüència de la radiació incident és més gran que un cert valor anomenat freqüència llindar  $\nu_0$ .
- Per a una radiació d'intensitat determinada, la intensitat del corrent augmentava fins a una intensitat de saturació, si augmentem el valor de potencial.
- Quan fem disminuir el valor de potencial fins a un valor negatiu, l'emissió fotoelectrònica disminueix fins a un valor fix, a partir del que ja no pot disminuir.

A principis del segle XX, Einstein va ser el primer a donar una explicació lògica al dit efecte, raonant que la llum estava formada per partícules anomenades fotons, que viatjaven a una velocitat  $c$  i que constituïen tots els altres tipus de radiació electromagnètica.

Aquest justificava la creació de corrent elèctric a partir de la llum amb l'impacte dels fotons que formaven dita radiació amb els àtoms del metall, provocant l'escapament dels seus electrons.

L'energia necessària per arrencar els electrons de la xarxa cristal·lina del metall va ser anomenada treball d'extracció  $W_0$

$$W_0 = h\nu_0$$

A partir d'aquests resultats, Einstein va crear l'equació de l'efecte fotoelèctric

$$h\nu = \frac{1}{2}mv^2 + h\nu_0$$

A més, es va concloure que la quantitat de moviment d'un fotó és

$$p = E/c = h\nu/c$$

### Dualitat ona-corpúscle:

L'any 1923, el físic francès Louis De Broglie, va generalitzar la dualitat entre els caràcters corpuscular i ondulatori de la llum a totes les partícules elementals existents. Tot seguit, va imposar la següent equació:

$$\lambda = h/mv = h/p$$

A més a més, l'expressió donada per De Broglie permetia conèixer el moment lineal dels fotons:

$$\lambda = h/\nu \Rightarrow p = E/c$$

### Principi d'incertesa:

A la mecànica quàntica, existeix un terme conegut com a funció d'ona, que ens permet conèixer la probabilitat de trobar una partícula qualsevol en una determinada regió de l'espai. Aquest terme s'expressava amb el símbol  $\psi$ , i va ser introduït per Erwin Schrödinger l'any 1926, com a conseqüència del seu model atòmic, en el que parlava d'ones de probabilitat.

Un altre dels aspectes fonamentals de la física quàntica és el principi d'incertesa, aquest diu: és impossible mesurar amb exactitud la posició i la quantitat de moviment d'una determinada partícula:

$$\Delta x \Delta p > h/2$$

on  $\Delta x$  és la incertesa en la posició i  $\Delta p$  és la incertesa en la quantitat de moviment.

### -Teoria de la Relativitat:

Es va publicar com a conseqüència del desig de mesurar la velocitat de la terra respecte a l'èter, desig molt present a finals del segle XIX.

A partir dels resultats negatius de l'experiment de Michelson-Morley es va veure que no hi havia moviment relatiu entre el moviment de la Terra i l'èter luminífer. Per tant es va arribar a les següents conclusions:

- La llum recorre els espais amb la mateixa velocitat  $c$ , a la que no es poden aplicar les transformacions de Galileu
- L'èter mai va existir, era una hipòtesi errònia

#### Els postulats de la teoria de la relativitat restringida:

Einstein, a part de per la demostració de l'efecte fotoelèctric, va ser molt reconegut per la creació de la Teoria de la Relativitat.

Aquesta, es pot separar en dues subteories: la teoria de la relativitat restringida, que estudia sistemes amb moviments uniformes, i la teoria de la relativitat general, que inclou l'estudi dels sistemes amb moviments accelerats.

En aquesta teoria, Einstein refuta l'existència de l'èter, com a conseqüència dels resultats obtinguts per Michelson i Morley l'any 1887.

Els seus postulats eren:

-Principi d'equivalència: Les lleis de la física es verifiquen de manera idèntica en tots els sistemes inercials, sense que sigui possible deduir a través seu cap distinció entre un sistema inercial i un altre.

-Segon postulat: La velocitat de la llum és una constant universal, invariant per a tots els observadors inercials. Aquest postulat, desencadenava diverses conseqüències:

-Que la velocitat  $c$  sigui independent de la velocitat relativa de l'observador, contradiu definitivament les lleis de Newton, i com a conseqüència, també té altres influències en l'espai i el temps. Així, si s'accepta dit postulat, s'observa que el temps és dependent de les característiques del sistema, és a dir, el temps és relatiu.

Aquesta expressió es pot matematitzar:

$$t = \frac{t'}{\sqrt{1-v^2/c^2}}$$

A més a més, aquesta dilatació temporal es fa present a la nostra vida quotidiana, tot i que és pràcticament imperceptible, com a conseqüència de l'ínfima relació de  $v$  i  $c$ .

#### Les transformacions Einstein-Lorentz: Contracció de la longitud:

Quan la velocitat  $v$  és molt més petita que la velocitat de la llum, es duen a terme dos processos: el de dilatació temporal, anteriorment esmentat, i el de dilatació longitudinal; aquest últim sempre es durà a terme en la direcció del moviment.

La denominada contracció Lorentz respondrà a la següent fórmula:

$$l = l_0 \sqrt{1 - v^2/c^2}$$

on  $l$  és la longitud obtinguda,  $l_0$  és la longitud impròpia

### Equivalència entre massa i energia:

La teoria de la relativitat, esmenta el fet que la massa d'un cos no és la mateixa per a tots els observadors de diferents sistemes, i que augmentava a mesura que adquiria certa velocitat:

$$m = m_0 / \sqrt{1 - v^2/c^2}$$

Així, s'expressa que cada massa existent a l'univers contenia una quantitat neta d'energia, responent a la següent expressió:

$$E = mc^2$$

Així, la combinació de la física quàntica i la teoria de la relativitat, juntament amb els estudis sobre el comportament de la radioactivitat, van conformar els pilars de la física actual, ajudant-nos a progressar més que mai al nostre coneixement sobre l'univers.

## Detector de partículas Timepix

### Introducció

El detector de partícules o chip Medipix/Timepix creat pel CERN és utilitzat, normalment amb aplicacions mèdiques, anàlisi de materials, òptica o, fins i tot, programes espacials. No obstant, un ajust del dispositiu permet que aquest sigui utilitzat per a l'estudi de partícules radioactives (com és el nostre cas).

El detector consta d'un detector de píxels montat pel chip Medipix (pixelat  $300\ \mu\text{m}$  Si,  $256 \times 256$  píxeles,  $55\ \mu\text{m}$  de to). A diferència dels detectors tradicionals com els tubs Geiger, el detector ofereix una visualització de les partícules en temps real, sent capaç de reconèixer els diferents tipus de partícules, mostrar pistes de partícules i funcionar com a espectòmetre (permet mesurar l'energia dipositada per cada partícula).

### Parts

El dispositiu consta diverses parts:

- Una tapa protectora (controlada amb una palanca)
- Connector USB
- Recobriments de l'electrònica amb forma rectangular
- Placa detectora de doble capa (composat per un sensor, un chip electrònic, metall, una capa de silici, un aïllant, un diode i un elèctrode de tensió de polarització)

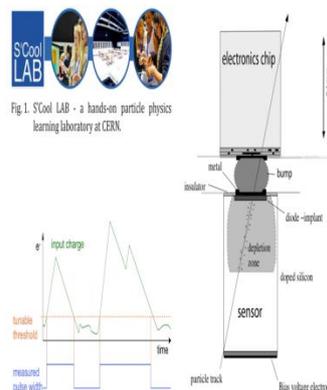


## Funcionament

### Funcionament de detectors de partícules híbrids

Els detectors de píxels híbrids tenen dues capes. La primera és un material pixelat, semi-conductor, on la radiació ionitzant que es transmet diposita determinades quantitats d'energia depenent del camí i l'impuls de les partícules mentre es recorre el material. Aquesta energia es pot mesurar mitjançant processos de ionització en el material semiconductor amb una matriu d'amplitud sensible a les càrregues. La informació podrà ser representada gràcies a un xip de lectura.

Aquest xip de lectura representa la segona capa i està segmentada en el mateix nombre de píxels. Les dues capes estan connectades una per una als píxels corresponents del sensor a través de solapaments de soldadura tal com es mostra a la imatge.



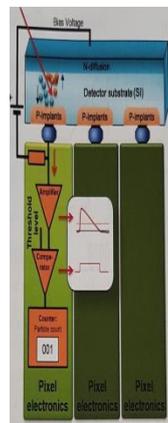
### Principi de detecció

L'energia que diposita la radiació ionitzant allibera parells de forats d'electrons a la zona d'esgotament d'un o diversos píxels en el material del sensor (fet de silici en el cas del nostre detector).

Els electrons o forats es recullen a través d'un camp elèctric aplicat externament, convertit en polsos de voltatge i digitalitzats per Timepix. (Aquí podria dir alguna cosa més, però entro dins de software).

## Electrònica

En aplicar una font de radiació, els electrons i els forats generats són atrets per electrodes que generen un pols de corrent elèctric. El pols de corrent elèctric que es crea després de l'impacte d'una partícula ionitzant es converteix en un pols de tensió i s'amplifica a l'amplificador. El pols amplificat continua amb una entrada d'un comparador que, en funció de la mida del pols, decideix comptar o ignorar l'impacte de la partícula. La segona entrada de comparador s'estableix en un nivell de separació definit anomenat llindar de detecció. Si el pols entrant és superior al nivell del llindar, es transmet un pols posterior a la sortida del comparador i es registra al comptador (mode comptador) o s'utilitza per determinar l'energia absorbida (mode espectròmetre). Un impuls inferior al nivell de llindar no es traslladarà a la sortida i serà ignorat. Establir el llindar pot limitar la detecció de pulsacions dèbils, que significa limitar el soroll, ja que poden donar-se senyals que no siguin polsos, però tindran un voltatge inferior. El nivell mínim de llindar correspon a la detecció d'energies de 3,5 keV en un píxel.



El càtode comú (a la part superior) està connectat a un potencial elèctric més alt que els ànodes del píxel inferior, el que significa que tots els díodes estan connectats en sentit invers. Cada píxel es connecta al xip de lectura mitjançant una bola de connexió.

L'operació del mode comptador i el mode espectròmetre es pot veure a la imatge. El gràfic de tensió en funció mostra tres polsos entrants marcats A, B i C, cadascun amb càrregues generades (és a dir, energia absorbida). El pols A es correspon amb la menor

energia absorbida i el pols C correspon a l'energia més alta absorbida. L'energia corresponent al pols A és baixa i, per tant, el pols no assoleix el nivell de llindar i no és registrat pel comptador. Els polsos B i C superen el llindar i cada un ha incrementat el valor del comptador en un.

En el mode espectròmetre, el detector assigna un número de cicles a cada pols basat en el principi ToT (temps sobre el llindar). D'aquesta manera, un oscil·lador envia un senyal amb una freqüència definida i el comptador compta quants cicles han passat no només a l'amplada del pols, sinó també a l'alçada d'aquest. Per tant, compta l'energia absorbida. Si realitzem un calibratge energètic del detector (comparant els polsos creats per fonts de radiació a la seva energia coneguda), podem (utilitzant una corba de calibratge) mostrar energies directament en kiloelectronvolts. Es pot veure el procés a la següent imatge:

