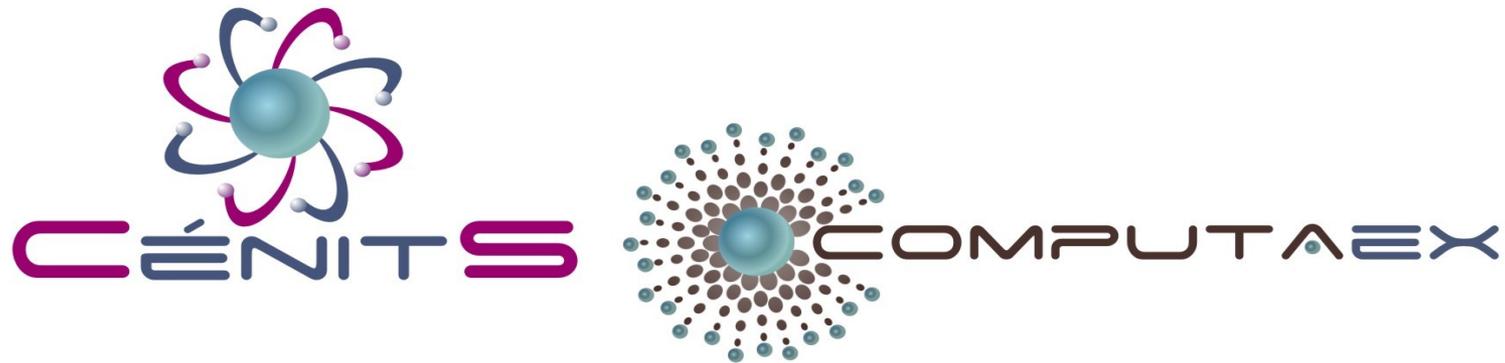


Lusitania

Innovando en Paralelo



César Gómez Martín
cesar.gomez@cenits.es
www.cenits.es

Esquema

- ¿Computación en Extremadura?
- ¿Qué es la programación paralela?
- ¿Por qué paralelizar?
- Ejemplo: Cálculo de números primos
- ¿Puedo paralelizar mi programa?
- Logros de la Supercomputación
- Supercomputación → Innovación

¿Computación en Extremadura?

- Los orígenes (1982)



```
(C) 1982 DRAGON DATA LTD  
16K BASIC INTERPRETER 1.0  
(C) 1982 BY MICROSOFT
```

```
OK
```

¿Computación en Extremadura?

- LUSITANIA (2009)



¿Computación en Extremadura?

- Comparativa

Dragon 32 (1982)	LUSITANIA (2009)
1 CPU (8 bits)	128 CPUs (doble núcleo) (64 bits)
Frecuencia CPU: 0,9 MHz	Frecuencia CPUs: 1,6 GHz
Memoria RAM: 64 KB	Memoria RAM: 2 TB
Almacenamiento: Casete	Almacenamiento: EVAs (~300 TB)

¿Cómo aprovechamos los nuevos computadores?

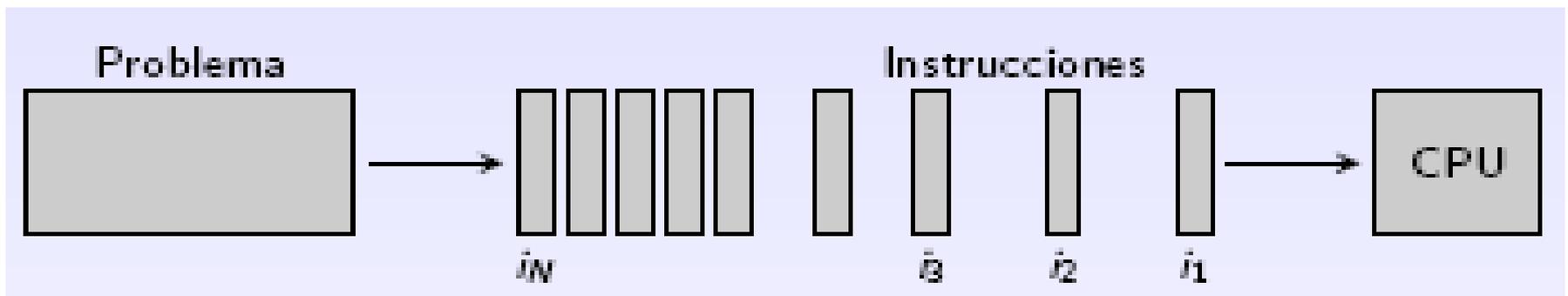
- Programando nuestras aplicaciones en paralelo

Estoy tranquilo,
totalmente tranquilo...



¿Qué es la programación paralela?

- Normalmente los programas realizan cálculos en serie:
 - Se ejecutan en un ordenador con una única CPU
 - Las instrucciones se ejecutan secuencialmente
 - Una única instrucción se ejecuta a la vez



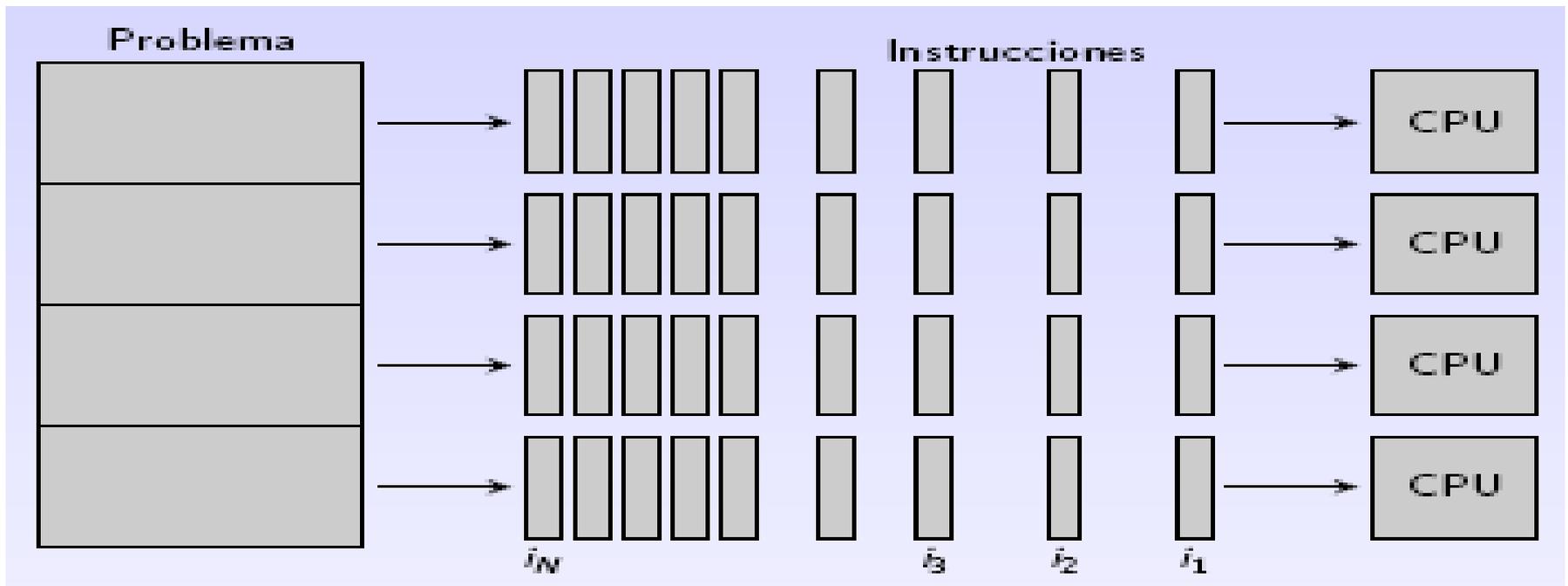
¿Qué es la programación paralela?

- La programación paralela consiste en usar varios recursos de forma simultánea para resolver un problema:
 - Se ejecutan en un ordenador con varias CPUs
 - El problema se divide en partes independientes
 - Cada parte se ejecuta simultáneamente



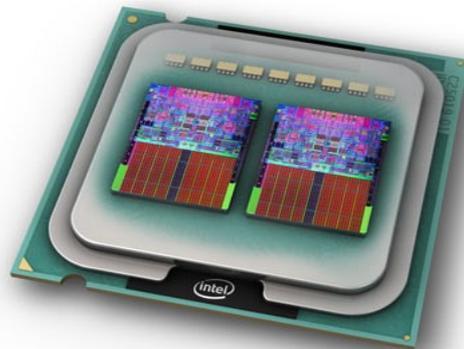
¿Qué es la programación paralela?

- La programación paralela consiste en usar varios recursos de forma simultánea para resolver un problema:



¿Por qué paralelizar?

- Resultados en menos tiempo (wall clock time)
- Solución a problemas mas grandes/complejos
- Posibilidad de realizar barridos paramétricos
- Estudio de diferentes variantes del problema
- Los procesadores actuales son de n-cores



Ejemplo: Cálculo de números primos

- Los sistemas de cifrado de clave pública aprovechan propiedades particulares de los números primos:
 - Es fácil multiplicar dos números primos juntos para obtener uno compuesto:
 - $1811 \times 3011 = \mathbf{5452921}$
 - Pero es difícil factorizar uno compuesto en sus componentes primos:
 - $??????? \times ??????? = \mathbf{3999944000147}$

Ejemplo: Cálculo de números primos

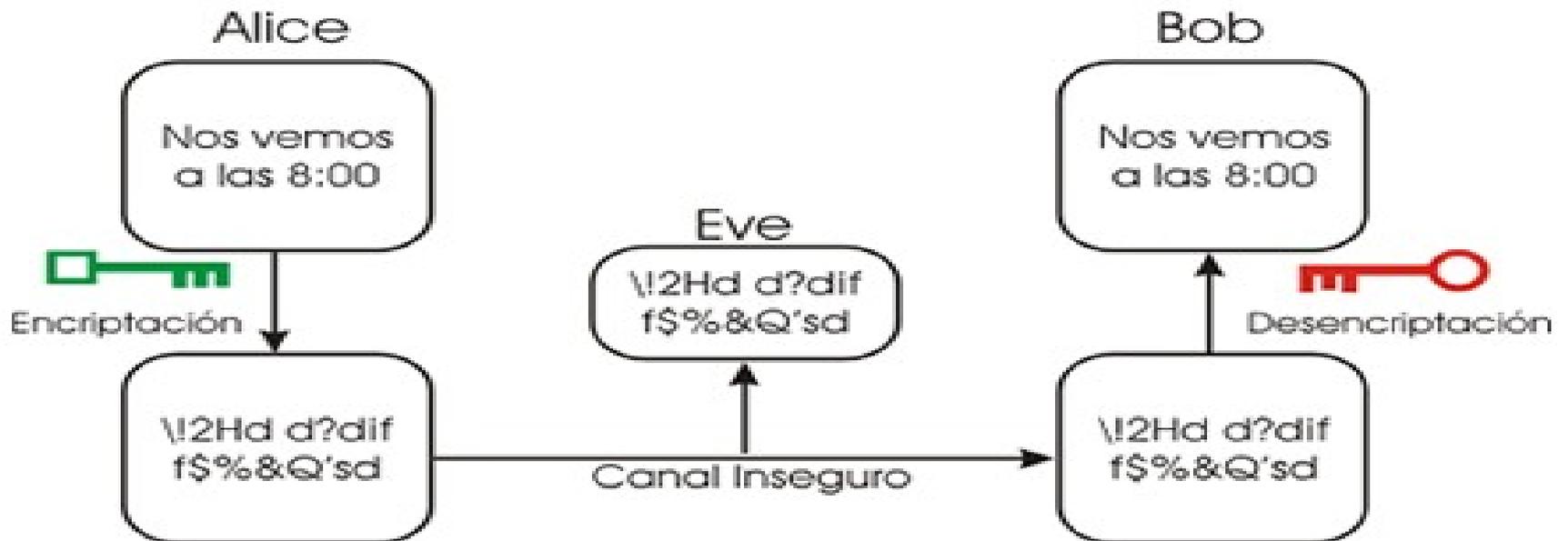
- Los sistemas de cifrado de clave pública aprovechan propiedades particulares de los números primos:
 - Es fácil multiplicar dos números primos juntos para obtener uno compuesto:
 - $1811 \times 3011 = \mathbf{5452921}$
 - Pero es difícil factorizar uno compuesto en sus componentes primos:
 - $??????? \times 1999993 = \mathbf{3999944000147}$

Ejemplo: Cálculo de números primos

- Los sistemas de cifrado de clave pública aprovechan propiedades particulares de los números primos:
 - Es fácil multiplicar dos números primos juntos para obtener uno compuesto:
 - $1811 \times 3011 = \mathbf{5452921}$
 - Pero es difícil factorizar uno compuesto en sus componentes primos:
 - $1999979 \times 1999993 = \mathbf{3999944000147}$

Ejemplo: Cálculo de números primos

Criptografía de Clave Pública



Clave Privada



Clave Pública

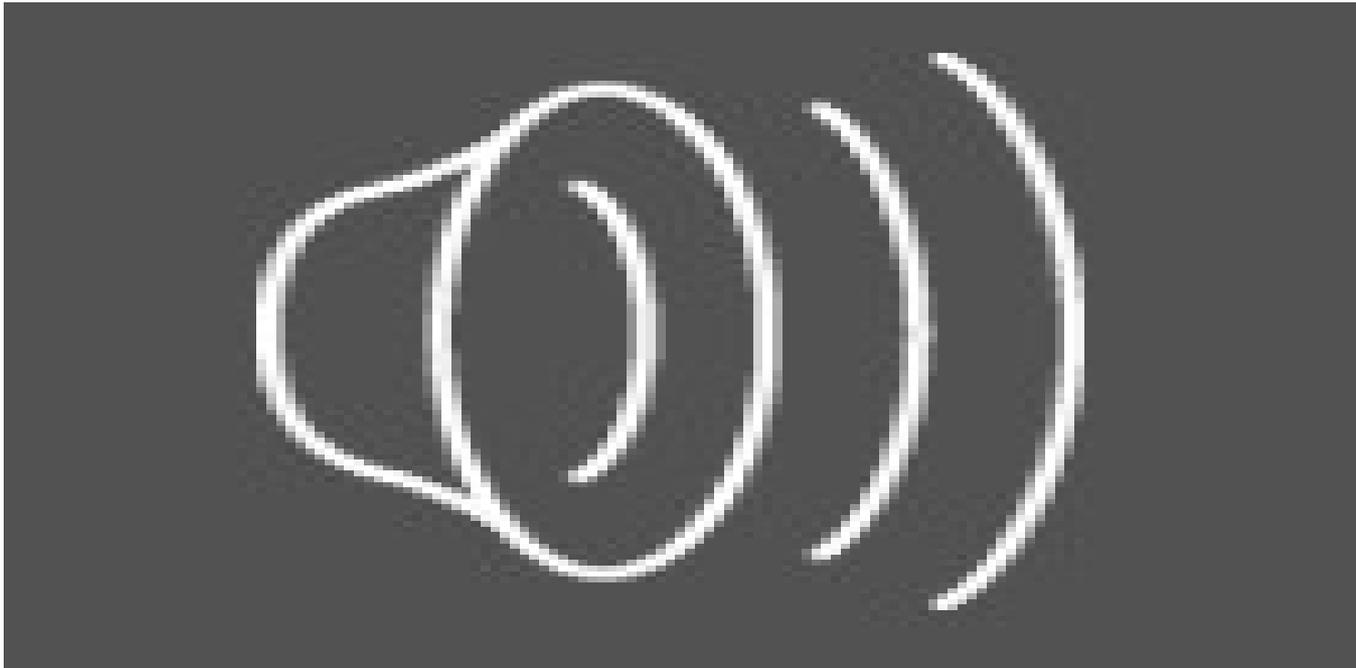
Ejemplo: Cálculo de números primos

- Calcular cuántos primos hay en los primeros 100.000.000 de números
 - Serie
 - Paralelo

	2	3	4	5	6	7	8	9	10	Prime numbers
11	12	13	14	15	16	17	18	19	20	
21	22	23	24	25	26	27	28	29	30	
31	32	33	34	35	36	37	38	39	40	
41	42	43	44	45	46	47	48	49	50	
51	52	53	54	55	56	57	58	59	60	
61	62	63	64	65	66	67	68	69	70	
71	72	73	74	75	76	77	78	79	80	
81	82	83	84	85	86	87	88	89	90	
91	92	93	94	95	96	97	98	99	100	
101	102	103	104	105	106	107	108	109	110	
111	112	113	114	115	116	117	118	119	120	

¿Puedo paralelizar mi programa?

- No se puede hacer una carrera de relevos en paralelo



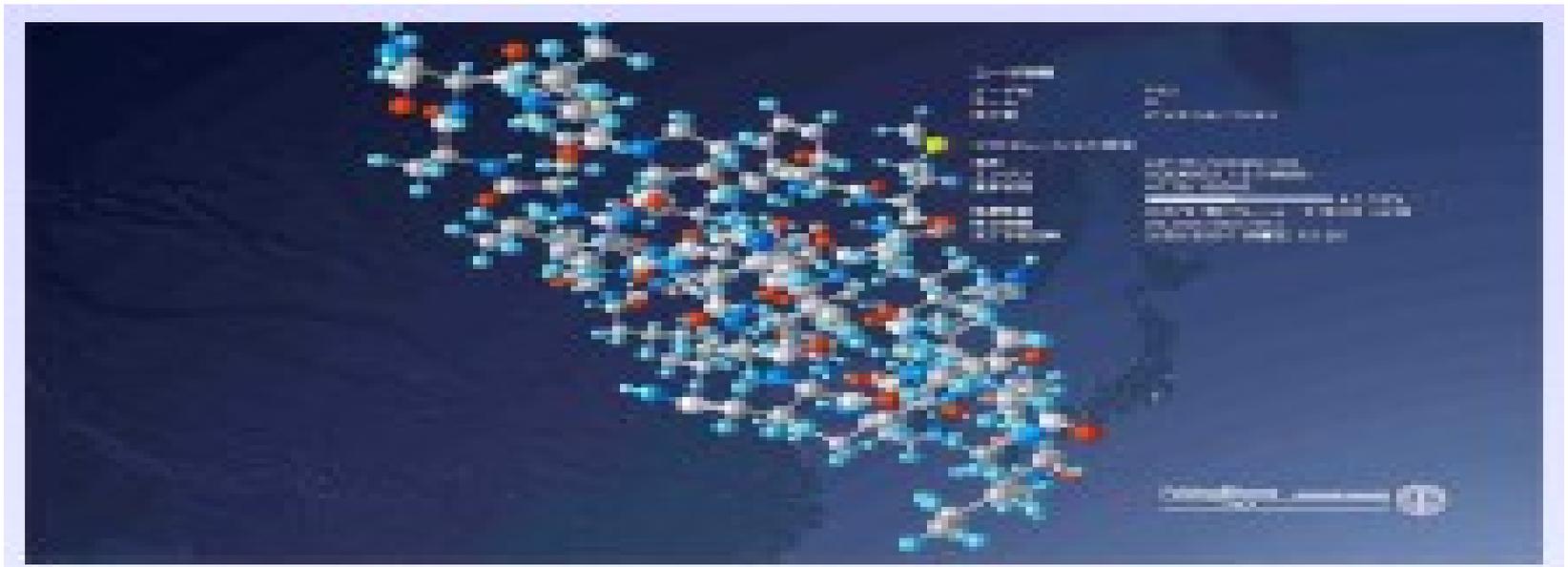
¿Puedo paralelizar mi programa?

- Pero se puede cocinar en paralelo



¿Puedo paralelizar mi programa?

- Hay problemas paralelizables y no paralelizables:
 - Problema paralelizable: calcular el potencial de energía de cientos de conformaciones posibles de una molécula



¿Puedo paralelizar mi programa?

- Hay problemas paralelizables y no paralelizables:
 - Problema **no** paralelizable: Cálculo de la serie de Fibonacci mediante su fórmula de recurrencia

$$F_{k+2} = F_{k+1} + F_k, \quad F_1 = 1, F_2 = 1$$

¿Puedo paralelizar mi programa?

- Ejemplos de aplicaciones paralelizables:
 - Conversión a grises de una imagen
 - La aplicación de un filtro a un píxel no depende de los píxeles vecinos



¿Puedo paralelizar mi programa?

- Ejemplos de aplicaciones paralelizables:
 - Cálculo de integral definida:

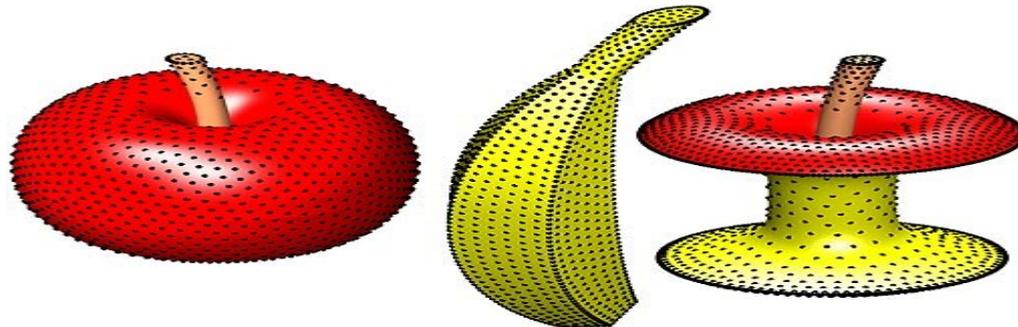
$$I = \int_a^b f(x) dx$$

- Se puede descomponer en:

$$I = \int_a^{x_1} f(x) dx + \int_{x_1}^{x_2} f(x) dx + \dots + \int_{x_N}^b f(x) dx$$

Logros de la Supercomputación

- Problema de los puntos de **Fekete**
 - Creación de moléculas estables
 - N^o 7 en la lista Smale:
 - Resuelto en 2 semanas en FinisTerra, hubiera tardado 40 años en un PC normal.



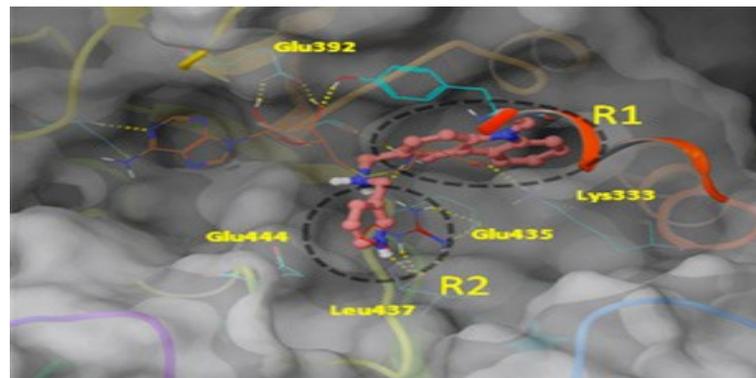
Logros de la Supercomputación

- Película **Avatar**
 - 40.000 procesadores
 - 104 TB de RAM
 - Coste 230 millones de dólares
 - Beneficio +2000 millones de dólares



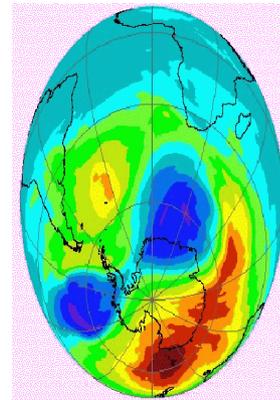
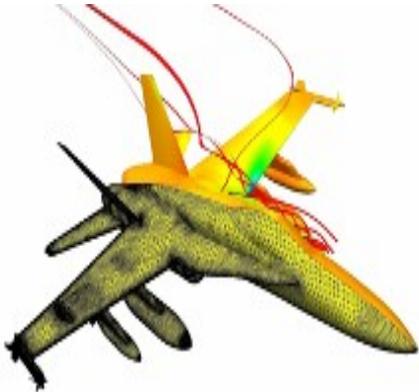
Logros de la Supercomputación

- Eliminación de **tumores cerebrales**
 - El biofísico Chenglong Li está creando simulaciones moleculares complejas para descubrir un medicamento que elimine tumores invasivos



Supercomputación → Innovación

- La Supercomputación es el tercer pilar de la ciencia del siglo XXI
 - Teoría
 - Experimentación
 - **Computación**

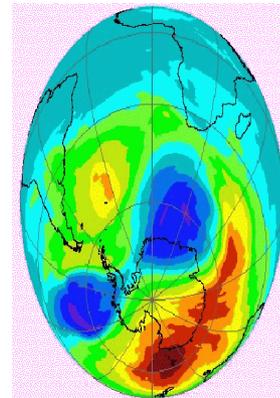
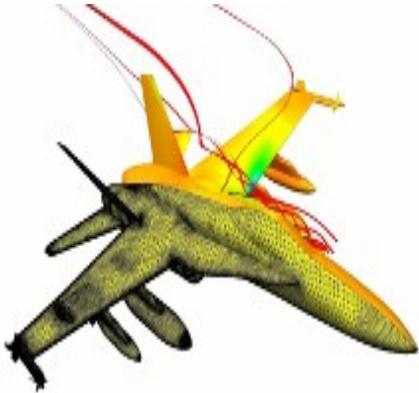


Supercomputación → Innovación

- Supongamos un problema en el que cuánto más rápido fuéramos, más rápido seríamos capaces de:
 - Eliminar una plaga que afecta a cultivos, animales o humanos
 - Encontrar la vía para combatir el calentamiento global
 - Construir un avión más rápido y ligero
 - Curar una enfermedad
 - Extender la vida de muchas personas

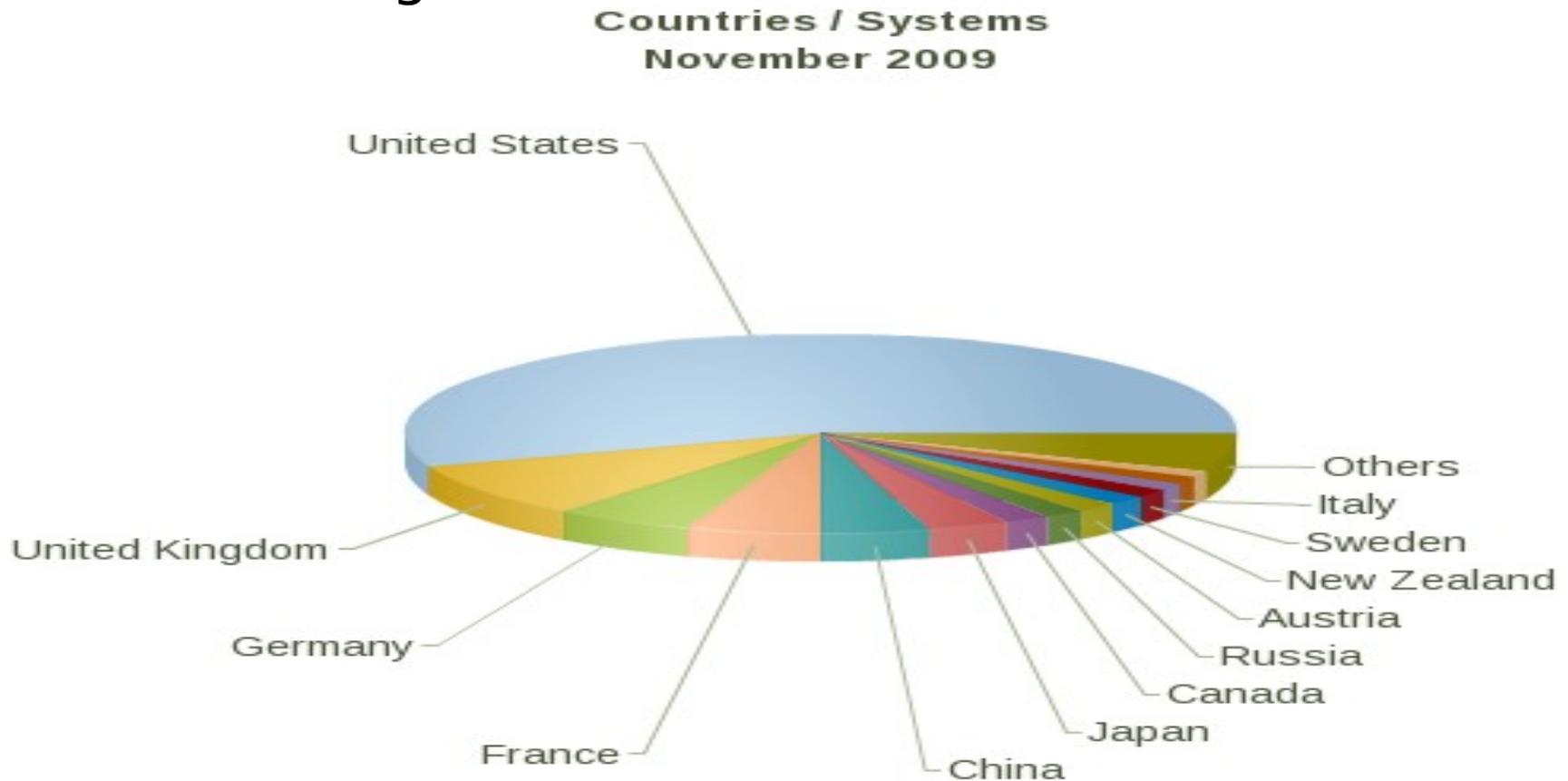
Supercomputación → Innovación

- La Supercomputación es un ingrediente clave en la capacidad de innovar, producir y crecer para ser competitivos.



Supercomputación → Innovación

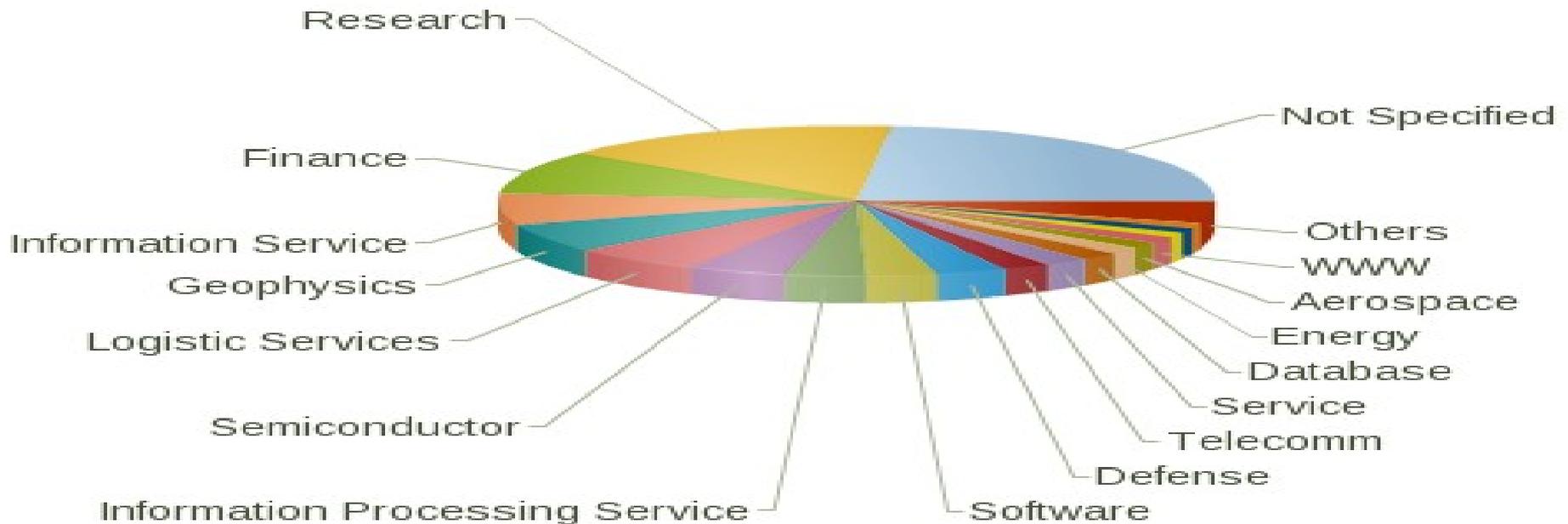
- TOP500.org



Supercomputación → Innovación

- TOP500.org

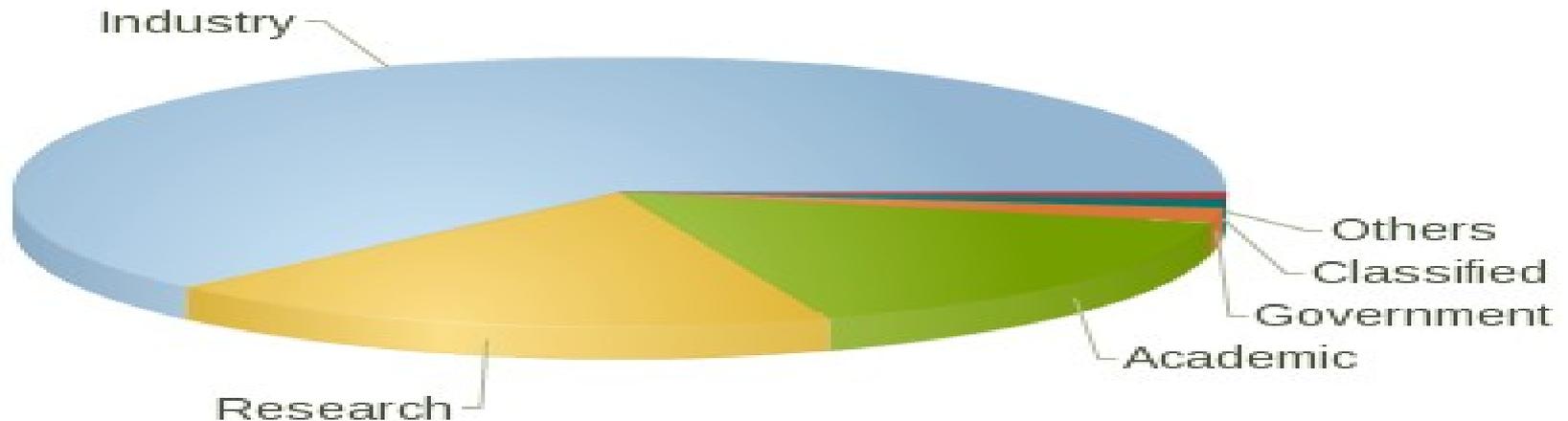
**Application Area / Systems
November 2009**



Supercomputación → Innovación

- TOP500.org

**Segments / Systems
November 2009**

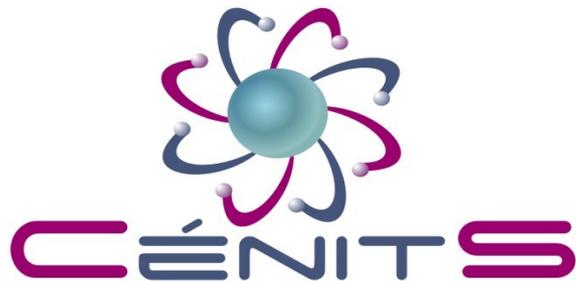


Supercomputación → Innovación

- Los supercomputadores:
 - Son la revolución del siglo XXI
 - Son capaces de resolver los problemas más complejos jamás imaginados
- Ser el primero no sólo determina quien obtiene conocimiento o riqueza, también puede permitir salvar la vida de muchas personas.



¡MUCHAS GRACIAS!



César Gómez Martín
cesar.gomez@cenits.es
www.cenits.es