

VISUALIZACION CIENTIFICA CON OPEN-DX

por Mario Storti

Centro Internacional de Métodos Numéricos

en Ingeniería - CIMEC

INTEC, (CONICET-UNL), Santa Fe, Argentina

mstorti@intec.unl.edu.ar

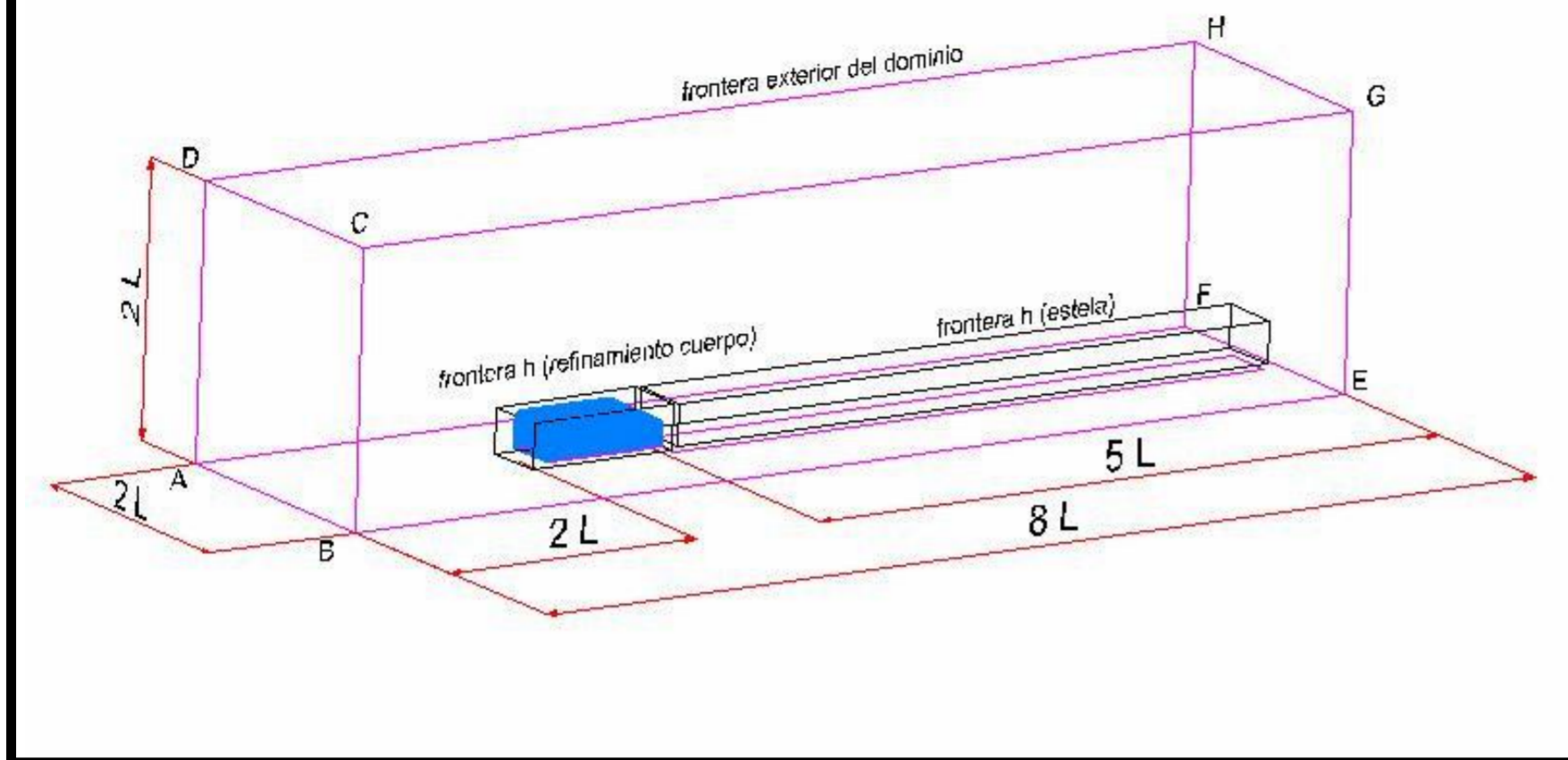
<http://venus.ceride.gov.ar/cimec>

29 de mayo de 2004

Que es la Mecánica Computacional?

- **Es una nueva disciplina científica que combina la matemática, física y la computación para resolver problemas científicos o de ingeniería.**
- **Los siguientes son ejemplos de aplicaciones de la mecánica computacional.**

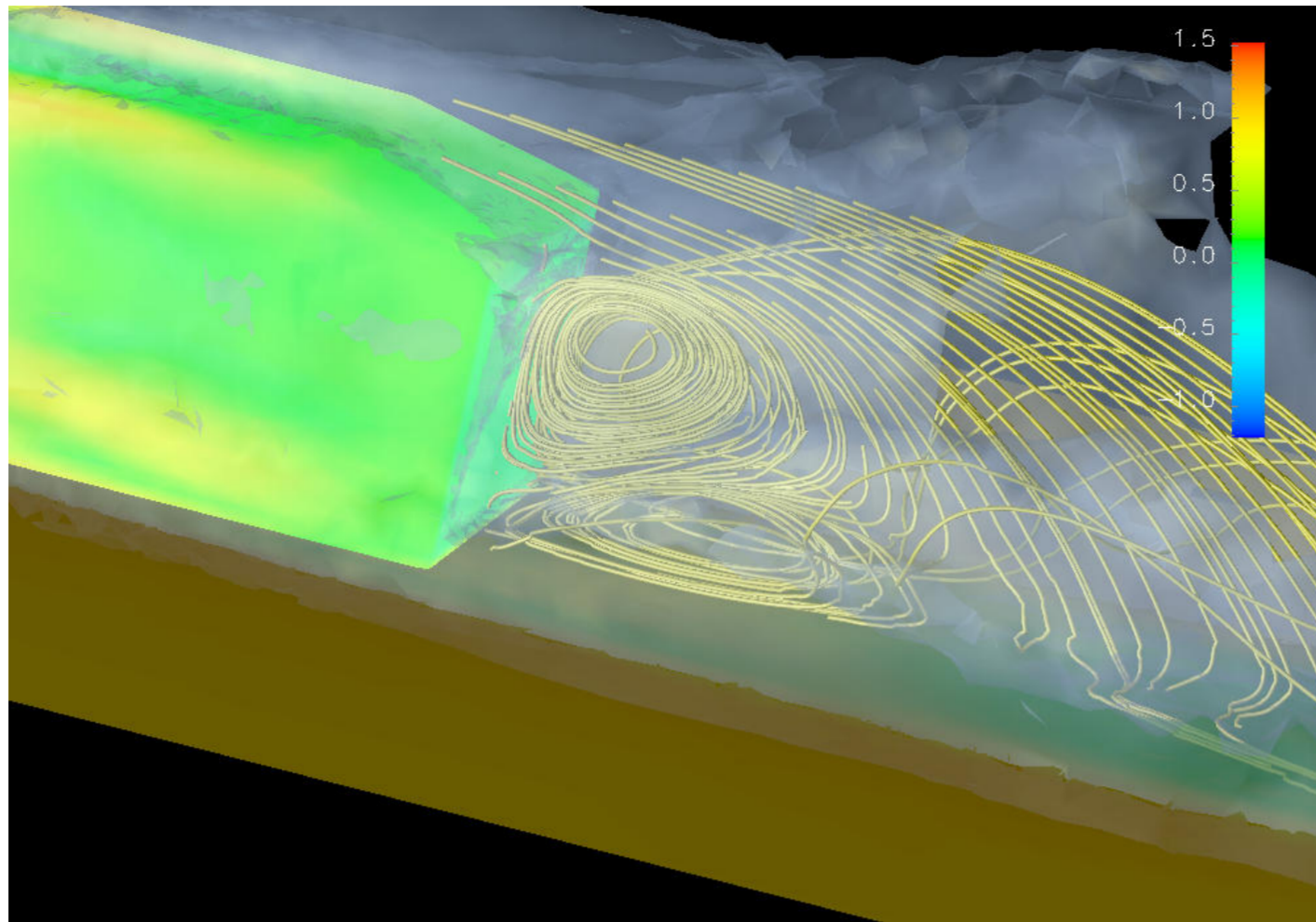
Aerodinámica exterior. Cuerpo de Ahmed



Aerodinámica exterior. Cuerpo de Ahmed (cont.)



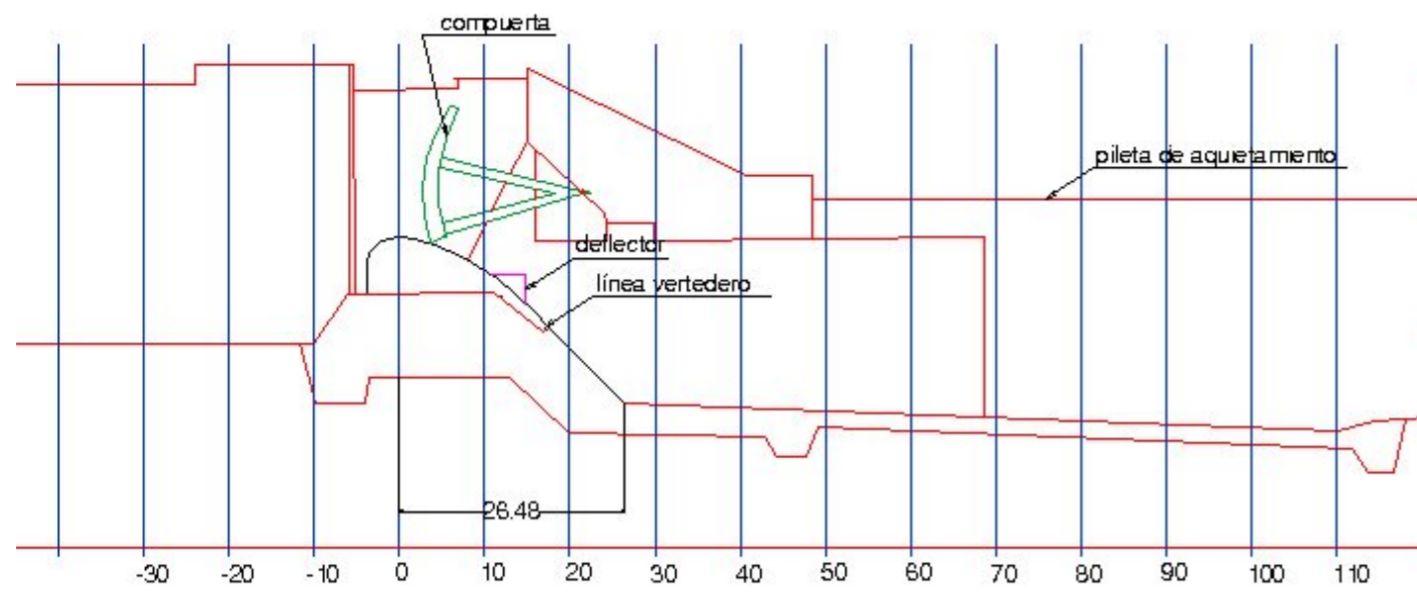
Aerodinámica exterior. Cuerpo de Ahmed (cont.)



Aerodinámica exterior. Cuerpo de Ahmed (cont.)

- [ahmed-circular-path.mpeg](#)
- [ahmed-rectilinear-path.mpeg](#)
- [ahmed-skin-rect-path-high.mpeg](#)
- [ahmed-skin-rect-path-low.mpeg](#)
- [ahmed-streamlines-x0.15.mpeg](#)
- [ahmed-streamlines-x0.mpeg](#)
- [ahmed-thin-stream.mpeg](#)
- [ahmed.mpeg](#)
- [image.mpeg](#)

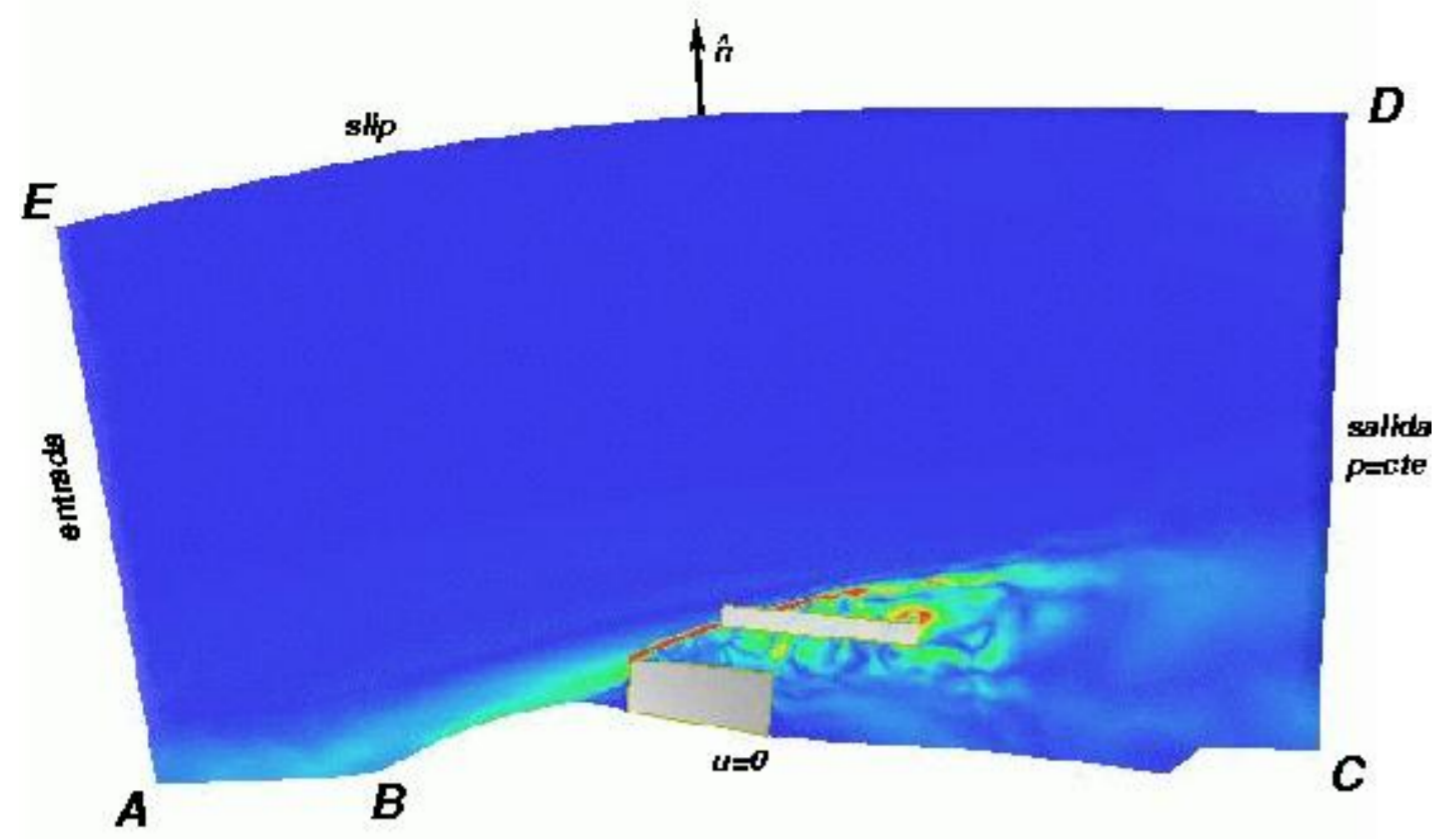
Estudio de impacto ambiental en Yacyreta.



Estudio de impacto ambiental en Yacyreta.

- [movie-6m-scan-reaction-N2.mpeg](#)
- [movie-6m-stream-reaction-N2.mpeg](#)
- [movie-6m-stream-reaction.mpeg](#)
- [spillwgc-6m-alpha-scan-plane.mpeg](#)

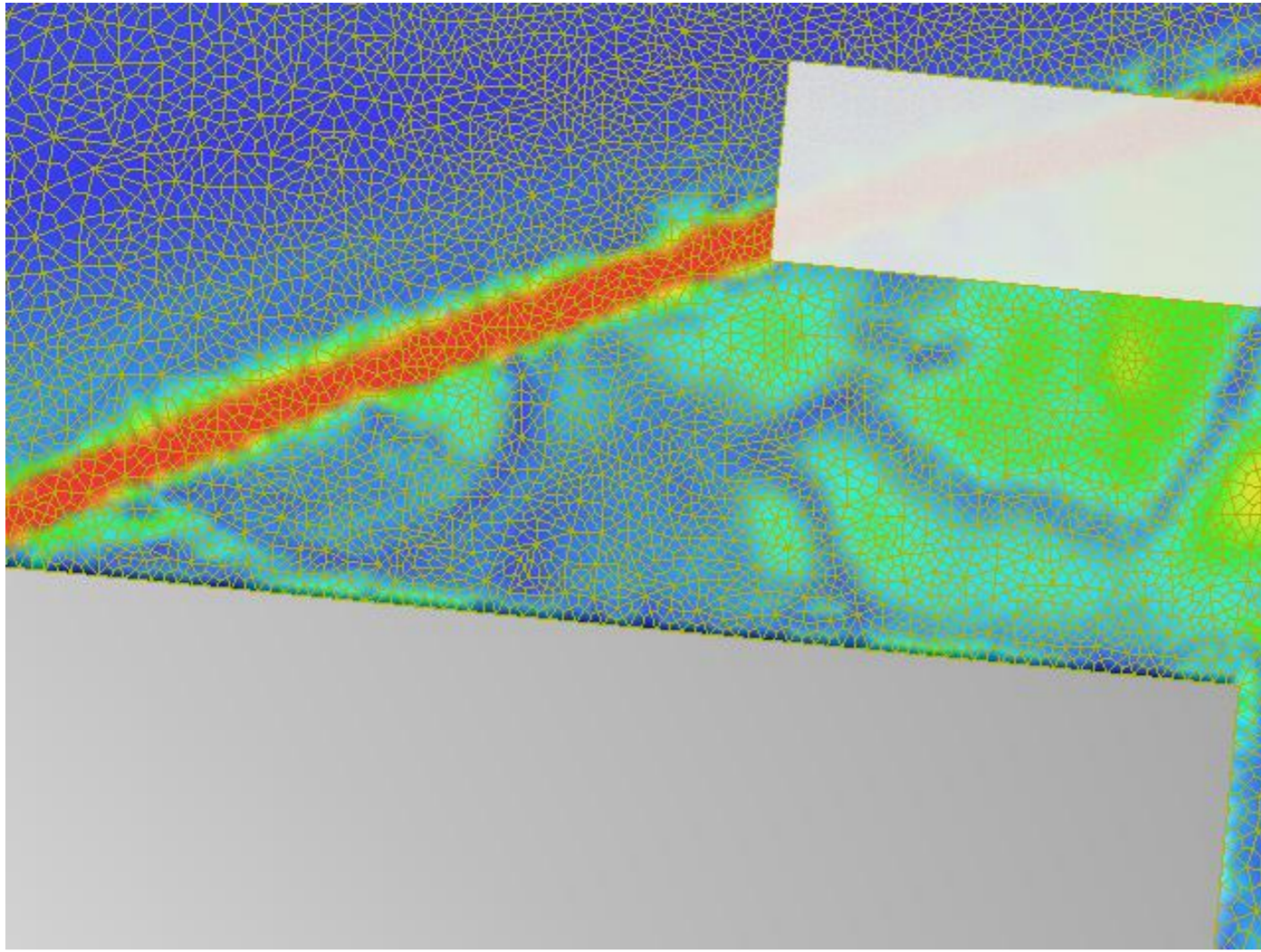
Impacto de obra civil en Puerto San Lorenzo. Rosario



Impacto de obra civil en Puerto San Lorenzo. Rosario (cont.)

- [absu-4m-2.mpeg](#)
- [absu-4m.mpeg](#)
- [isosurf-vort-4m.mpeg](#)
- [isosurf-vort.mpeg](#)
- [movie-H5-4m-absu.mpeg](#)
- [movie-H6-4m-absu.mpeg](#)
- [movie-JPG-cl-z7m-vort.mpeg](#)
- [muelle.mpeg](#)
- [vector-z6.mpeg](#)
- [vort-4m.mpeg](#)
- [vort-fs-7m.mpeg](#)
- [vort-stream-4m.mpeg](#)
- [vort-stream.mpeg](#)

Impacto de obra civil en Puerto San Lorenzo. Rosario



Fuerzas sobre un transporte de líquidos

- [tank.mpeg](#)
- [tank2.mpeg](#)
- [tank3.mpeg](#)

Cavidad cúbica

- [cubcav-b5000.mpeg](#)

Cavidad cuadrada

- [movie_sqcav4.gif](#)
- [movie_sqcav5.gif](#)

Simulación experimental

- Para cada una de las formas propuestas crear un modelo y medir en laboratorio la perturbación en el muelle.
- Es un procedimiento costoso.
- En ciertos casos no es factible, por razones económicas, construir un modelo a escala completa sino que se debe recurrir a modelos a escala reducida, introduciendo un error en la estimación.
- En algunos problemas no es posible medir en forma directa: por ejemplo medir el campo de velocidades en un proceso siderúrgico con acero líquido a 1600°C .

Simulación numérica

- Resolver las ecuaciones que gobiernan el fluido (Ecuaciones de Navier-Stokes/Euler/Flujo Potencial).
- Dividir el dominio en pequeños “elementos”.
- Asumir variaciones simples de las variables dentro de cada elemento.
- Esto lleva a sistemas de ecuaciones lineales con tantas incógnitas como grados nodos por campos incógnita existen.
- No confundir simulación numérica con los (hoy muy comunes) efectos especiales usados en películas, etc...!!
- La simulación numérica es uno de las disciplinas relacionadas con la computación más antiguas. De hecho, uno de los principales usos de las primeras computadoras fue la simulación numérica en la industria de la aviación, nuclear, etc...

Ecs. de Navier-Stokes en una cavidad cúbica. (cont.)

- Resolver el movimiento de un fluido dentro de un cubo.
- Dividimos cada una de las dimensiones en 100 segmentos.
- Se originan $100 \times 100 \times 100 = 1.000.000$ de pequeños cubitos (elementos)
- Por cada elemento hay cuatro incógnitas: 3 componentes de velocidad y la presión: $N = 4.000.000$ de incógnitas. Todas las operaciones se hacen en doble precisión.
- El sistema a resolver tiene una matriz de
 $N \times N = 1.6 \times 10^{13} = 1.28 \times 10^{14}$ bytes = 128 Tbytes elementos.
Afortunadamente casi todos los elementos de cada línea de esta matriz son cero, y sólo debemos guardar aquellos que no lo son. Hay
 $N_{vec} = 4 \times 27 = 108$ coeficientes no nulos en cada línea de la matriz lo cual hace un total de $N \times N_{vec} = 4 \times 10^6 \times 108 = 3.4$ Gb

Ecs. de Navier-Stokes en una cavidad cúbica (cont.)

- Factorización de la matriz requiere

$O(Nm^2) = O(4 \times 10^6 \times (4 \times 10^4)^2) = 6.4 \times 10^{15}$ operaciones de punto flotante. m es el número de grados de libertad que hay en una "capa" 2D de nodos ($m = 4 \times 100 \times 100$). A una velocidad de procesamiento de 2.4 Gflops (2.4×10^9 operaciones de punto flotante por segundo p.ej. Pentium III 2.4 Ghz). Esto tardaría $6.4 \times 10^{15} / 2.4 \times 10^9 / 3600 = 740$ hrs!!. En realidad la velocidad de procesamiento es menor porque la velocidad de procesamiento está limitada por la velocidad con la que los datos pueden fluir desde la memoria RAM al procesador y la velocidad de procesamiento resulta en 300 Mflops. Esto lleva la evaluación del tiempo de procesamiento a ~ 6.000 hrs!!

Ventajas

- El costo de las simulaciones numéricas es menor que las simulaciones experimentales y tiende a disminuir constantemente.
- Permite conocer el estado del fluido en todo punto del dominio.
- No hay problemas de escala o peligrosidad.
- Los mismos recursos (hardware y software) pueden ser usados para una variedad de problemas de ingeniería.

Desventajas

- El modelo matemático a resolver no es perfectamente conocido para ciertos problemas.
- El grado de refinamiento a veces no es suficiente para obtener una representación adecuada.

Técnicas de computación de alta-performance (HPC)

- En la búsqueda de modelizaciones cada vez más precisas, se van desarrollando técnicas para incrementar la capacidad de cálculo.
- Una de las técnicas más comunes y prometedoras es la de dividir el problema en subproblemas más pequeños y resolver cada uno de los problemas en un procesador por separado. A esto se le da el nombre de *“procesamiento distribuido”*.
- En el mejor de los casos, si cada uno de los subproblemas es independiente del otro (están *“desacoplados”*) entonces el tiempo de cálculo en n procesadores es $T_n = T_1/n$.
- Debido a que, en general, los problemas no están desacoplados es necesario pasar cierta información de un procesador a otro. Es decir que $T_n = T_1/n + T_{\text{comm}}$.

Evolución del hardware usado en el procesamiento distribuido

- En el extremo “superior” están las grandes firmas que venden supercomputadoras como Cray (hoy SGI) o IBM Ascii-Red con miles de procesadores. Hoy las más comunes son las SGI Origin 2000.
- NOW/COW: A partir de los 80's era común encontrar en los laboratorios de las universidades un cierto número de estaciones de trabajo (SUN/HP/DEC/SGI) y surgió la motivación de utilizar esta potencia de cálculo en corridas nocturnas. Surgieron las bibliotecas de “Paso de Mensajes” como PVM y MPI.

Cray T94. CSC/UFRGS - Porto Alegre Brasil

- Cray T94 en Centro de Supercomputación UFRGS, Porto Alegre Brasil.
 - 2 Procs. × 1.8 Gflops, Costo aprox. \$ 5,000,000.
- <http://www.cesup.yfrgs.br>



Clusters Beowulf

- Con el abaratamiento de las PC's y el advenimiento de software libre surgió la posibilidad de crear clusters de PC's completamente dedicados a cálculo. Estos son los llamados clusters Beowulf.
- De *"How to build a Beowulf"* (Sterling, T.L. et.al., MIT Press, 1999) un *"cluster Beowulf"* es *"Un cluster the 'mass-market commodity off-the-shelf' (M2COTS) PC's interconectadas por tecnología LAN de bajo costo corriendo un OS open source de tipo Unix y ejecutando aplicaciones en paralelo con una librería de paso de mensajes estándar en la industria."* El *"Proyecto Beowulf"* fue desarrollado originalmente en el *Goddard Space Flight Center (GSFC)*. También son populares los clusters con procesadores DEC/Alpha.

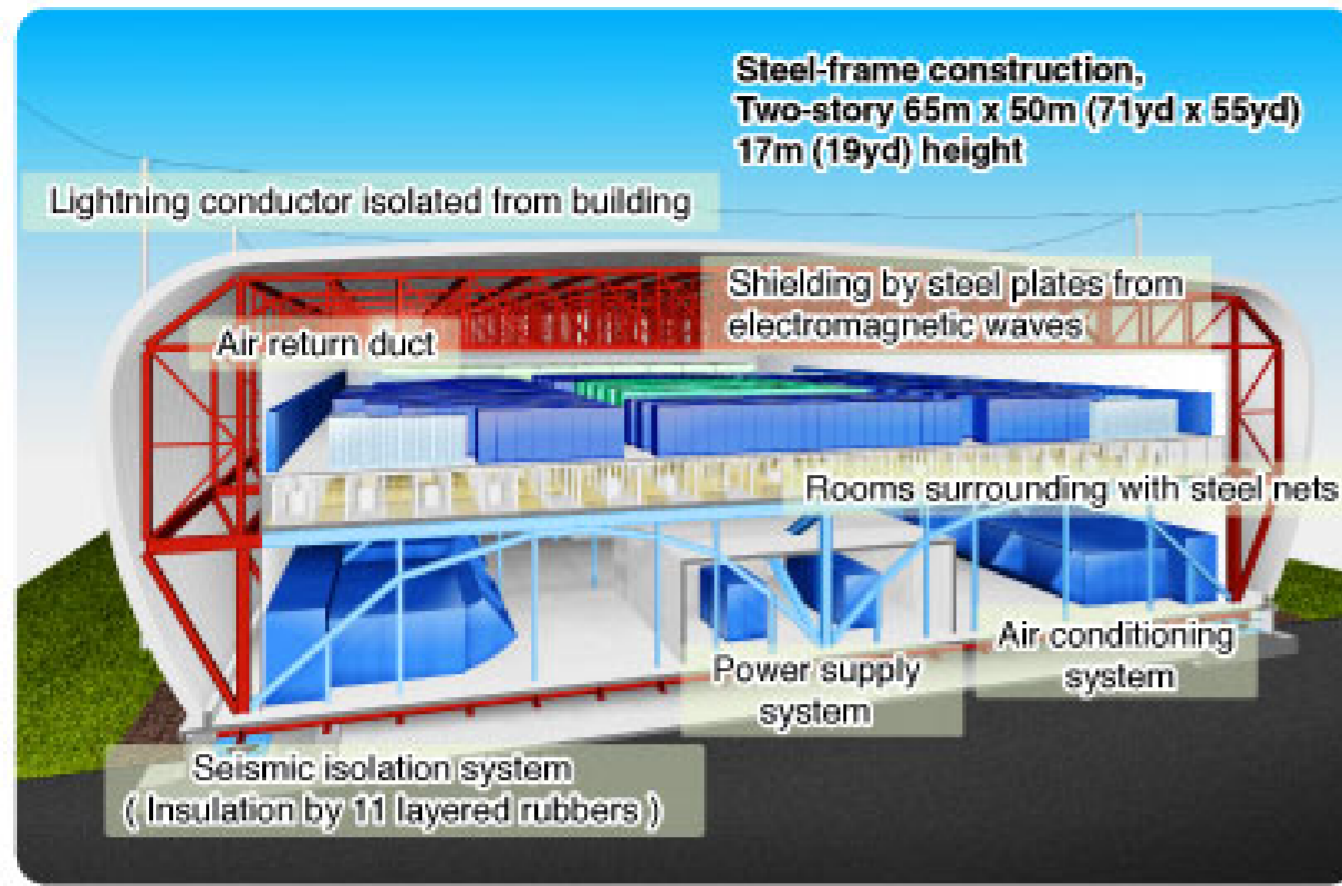
Top 10 list

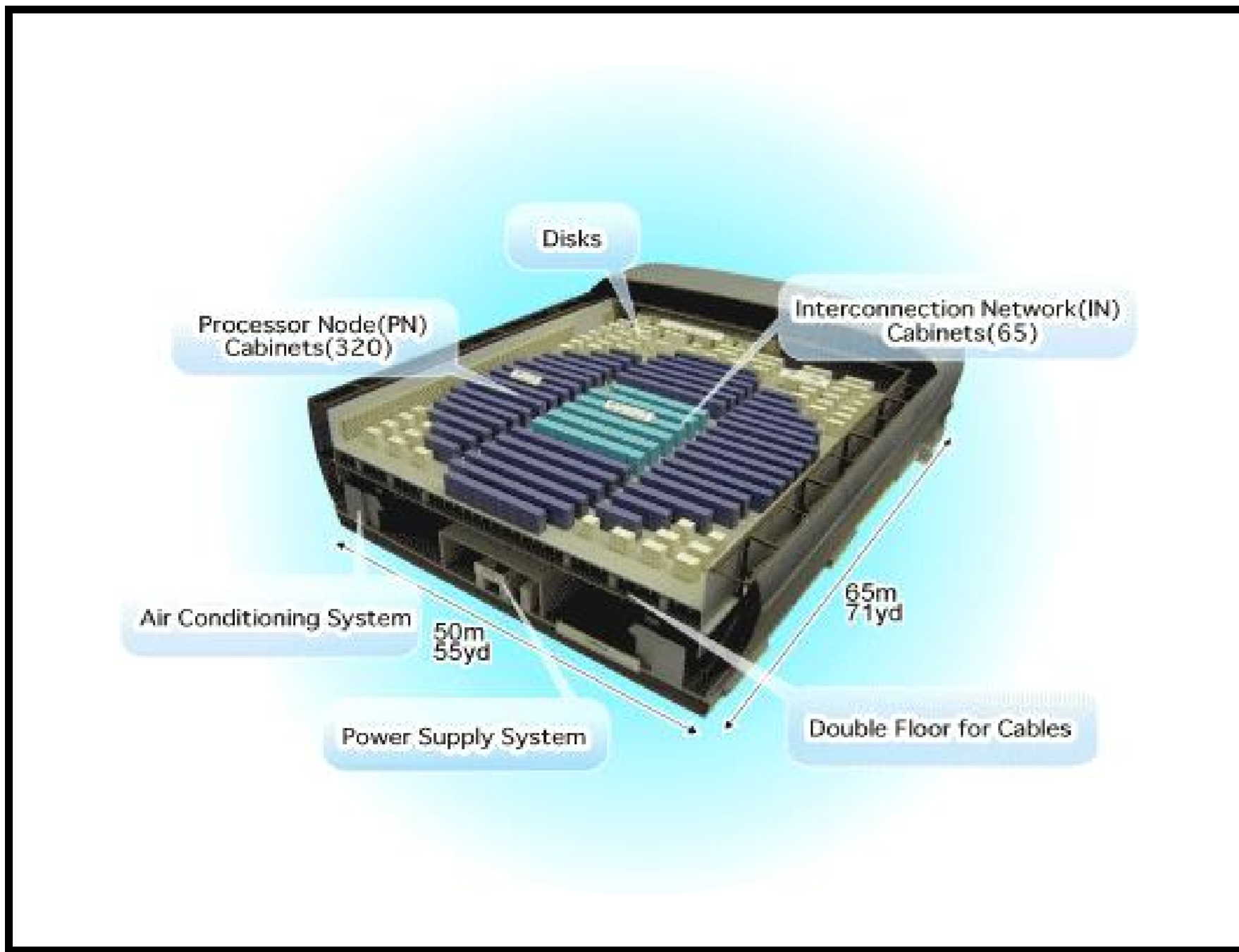
(sacado de www.top500.org)

- The Earth Simulator, built by NEC, remains the unchallenged #1 at 40 Tflops. (4e12 flop/sc)
- ASCI Q at Los Alamos is still #2 at 13.88 TFlop/s.
- The third system ever to exceed the 10 TFlop/s mark is Virginia Tech's X measured at 10.28 TFlop/s. This cluster is built with the Apple G5 as building blocks and is often referred to as the 'SuperMac' in media reports. It uses a Mellanox network based on the new Infinband technology as interconnect.
- The fourth system is also a cluster. The Tungsten cluster at NCSA is a Dell PowerEdge-based system using a Myrinet interconnect. It just missed the 10 TFlop/s mark with a measured 9.82 TFlop/s.
- The list of clusters in the TOP10 continues with the upgraded Itanium2-based Hewlett-Packard system, located at DOE's Pacific Northwest National Laboratory, which uses a Quadrics interconnect.

Earth simulator (1st pos.)

- **5,120 (640 8-way nodes) 500 MHz NEC CPUs**
- **8 GFLOPS per CPU (41 TFLOPS total)**
- **2 GB (4 512 MB FPLRAM modules) per CPU (10 TB total)**
- **shared memory inside the node**
- **640 640 crossbar switch between the nodes**
- **16 GB/s inter-node bandwidth**
- **20 kVA power consumption per node**





Necesidad de la vis. científica

- Los grandes volúmenes de datos generados obliga a la necesidad de técnicas especiales para la visualización.
- Sobre todo en 3D, por ejemplo visualización de patrones de flujo es particularmente difícil.
 - Se separa el flujo?
 - Espesor de capa límite.
 - Intensidad turbulencia.
 - Vorticidad.
 - Esfuerzos de corte.

Vis. científica en otras áreas

- **Datos de mediciones**
- **Medicina**
- **Meteorología**

Evolución de la visualización en mec. comp.

- Impresiones con caracteres, ver los números, gráficos xy
- Estaciones gráficas, graficos 2D. Isocurvas. Gráficos en colores.
- Visualización 3D animaciones

Algunos paquetes de visualización científica

- **Gid (www.cimne.upc.es)** Orientado a elementos finitos. Generación de mallas. Postprocesamiento.
- **VTK Visualization Toolkit.** Librería de visualización científica en C++. Requiere escribir programas para la visualización.
- **Paquetes comerciales: Matlab, IDL...**

OpenDX

- Originariamente era el IBM Data Explorer, producido y comercializado por IBM
- Recientemente IBM liberó el código y un grupo de colaboradores dentro de la comunidad de OpenSource se encarga de su mantenimiento.
www.opendx.org
- Se puede obtener soporte de listas de correo y en forma comercial de Viz. Inc

Instalación de DX

- Requiere OpenGL
- Librería Motif (No lo pude hacer andar con Lesstif. Anda OK con OpenMotif)
- Existen paquetes RPM.
- Básicamente instala
 - Una serie de binarios (dxexec, dxui, dx, dxworker...)
 - Librerías (libDX.a, libDXL.a, libDXlite.a ...) y headers (dx.h...)
 - Documentación, ejemplos...

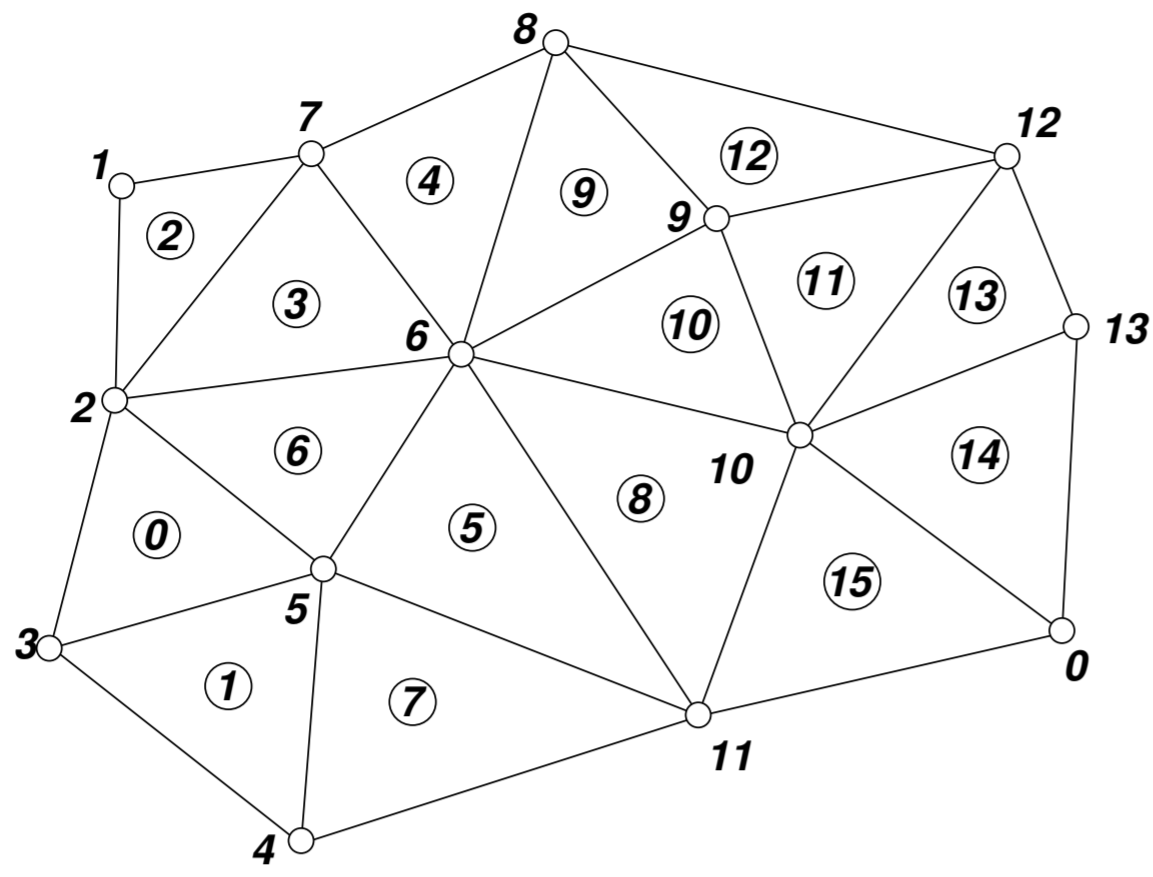
Posibilidades de DX

- **Permite editar programas gráficos con un Visual Program Editor (Editor de Programas Visual)**
- **Los programas gráficos pueden obtener datos de usuario a través de paneles de control (tamaño de vectores, mapas de colores, ancho de líneas de corriente, densidad de puntos...)**
- **También se pueden editar programas en forma de scripting.**
- **Se pueden escribir módulos en C/C++ que se linkean dinámicamente con DX.**
- **Se pueden escribir aplicaciones independientes que se comunican con DX vía un protocolo de paso de mensajes (librería DXLink).**
- **Se pueden escribir aplicaciones de visualización ad-hoc para ciertos problemas escribiendo código en C/C++ y linkeando con las librerías de DX (ejecutables que contienen a DX).**

Conceptos básicos

- El objeto básico para la visualización con DX es el “campo” (“field”)
- Un campo tiene 3 atributos básicos:
 - Posiciones: (“positions”) arreglo de $N \times \text{ndim}$ floats (N puntos en (2D/3D))
 - Conectividades: (“connections”) arreglo de $N_{\text{elem}} \times n_{\text{el}}$ enteros. Hay sólo cuatro tipo de conectividades posibles
 - $n_{\text{el}}=3, \text{ndim}=2$ ó 3 “triangles”
 - $n_{\text{el}}=4, \text{ndim}=2$ ó 3 “quads”
 - $n_{\text{el}}=4, \text{ndim}=3$ “tetrahedra”
 - $n_{\text{el}}=8, \text{ndim}=3$ “cubes”
 - Datos: (“data”) arreglo de $N_{\text{data}} \times \text{ndof}$ floats que representan los datos físicos a visualizar. N_{data} puede ser igual al número de puntos N (“position-dependent”) o igual al número de conexiones N_{elem} (“connection-dependent”).

Conectividades de triángulos



PETSc-FEM y DX

- **PETSc-FEM es un programa de elementos finitos desarrollado en el CIMEC <http://www.cimec.org.ar/petscfem>**
- **Escrito en C++ con filosofía orientada a objetos.**
- **Implementado en paralelo usando PETSc/MPI**
- **Navier-Stokes incompresible, compresible, shallow water, advección-difusión, flujo en medios porosos, elasticidad lineal.**

PETSc-FEM y DX (cont.)

- La mayoría de los ejemplos mostrados fueron obtenidos con PETSc-FEM.
- PETSc-FEM dialoga con DX vía un módulo escrito en C++ linkeado dinámicamente con DX y que intercambia información via sockets.
 - Crear archivos de datos (.dx) con scripts
 - Lanzar PETSc-FEM para leer resultados previos y visualizar con DX.
 - Lanzar PETSc-FEM para visualizar resultados en tiempo de ejecución.

Animaciones. How-to actual

- **Generamos secuencias de vistas con un sequencer.**
- **Frecuentemente se interpolan de 2 a 6 cuadros por paso de tiempo interpolando linealmente los estados.**
- **DX genera archivos .tiff**
- **Con una llamada a “System” se convierte el .tiff a .yuv, se compacta y se almacena en un directorio.**
- **Se generan videos con formato MPEG con 400 a 600 frames, a 16-20 fps. Se utiliza ffmpeg (streaming video encoder).**