

Introducción

Visualización, Discretización, Hardware y Métodos Numéricos

Contenido

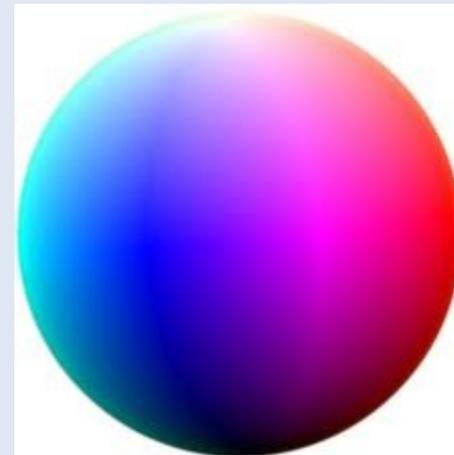
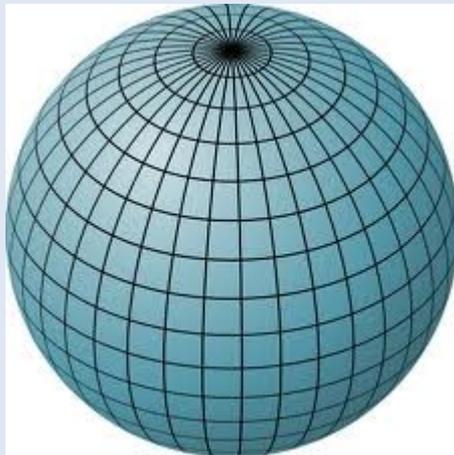
- Visualización científica versus Visualización realista
- Modelos discretos
 - Discretización de objetos
 - Discretización en hardware gráfico
 - Discretización y métodos numéricos

Visualización científica versus Visualización Realista

- **Visualización científica:** técnicas de visualización para ilustrar gráficamente datos científicos de tal manera de facilitar la comprensión de éstos (grandes volúmenes de datos)
 - Ejemplos en: http://en.wikipedia.org/wiki/Scientific_visualization
- **Visualización realista:** técnicas para visualizar escenas generadas por el computador de modo que parezcan reales
 - Ejemplos en: http://en.wikipedia.org/wiki/Fractal_landscape

Ámbitos de uso de visualización científica

- Datos capturados/medidos por instrumentos/satélites: datos meteorológicos, terrestres, astronómicos, etc.
- Datos generados por simulaciones computacionales
- Funciones matemáticas explícitas
 - Ejemplo: Visualizar una esfera $x^2+y^2+z^2 = r^2$

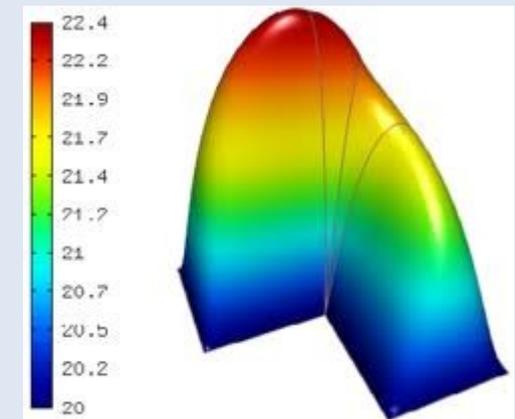
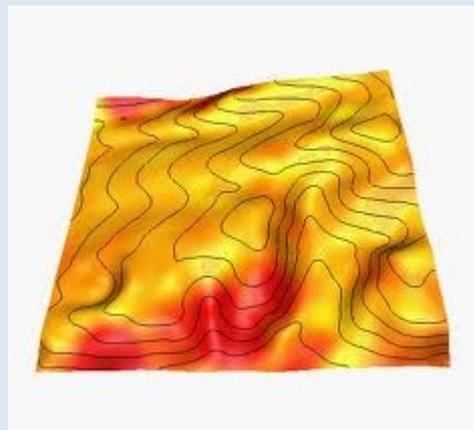
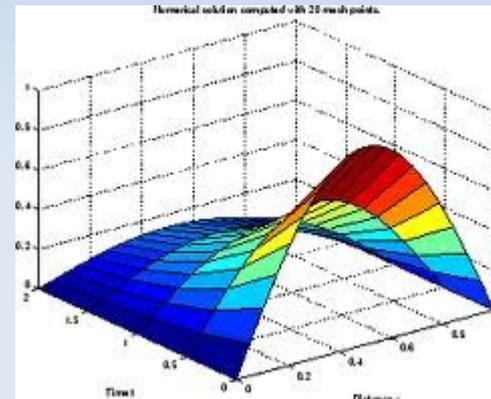
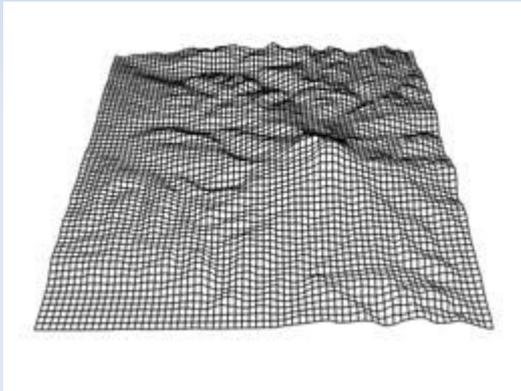


Ámbitos de uso de visualización científica

- Aplicaciones en:
 - Ciencias naturales: formación de estrellas, moléculas, etc
 - Geografía y Ecología: representación de la topografía, simulación del clima, contaminación atmosférica, etc
 - Ciencia aplicadas: diseño de aviones, visualización de ciudades, diseño de semiconductores, etc.
 - Geología y minería: visualización de sitios de exploración minera, yacimientos explotados, etc.
 - Biología y medicina: apoyo a la cirugía, estudio de estructuras biológicas

Visualización científica: Problemas básicos

- Visualizar un terreno para x,y en un cuadrado
- Visualizar la solución de una EDP en un dominio 2D

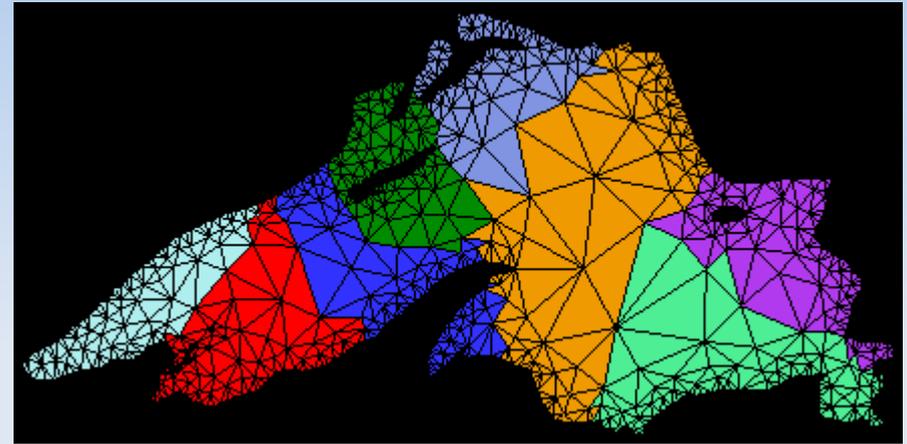


Visualización científica: Problemas básicos

- Qué tiene en común la visualización de los problemas anteriores? Es fácil? En qué se diferencian?

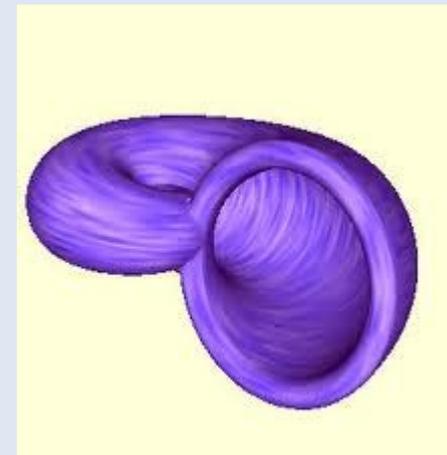
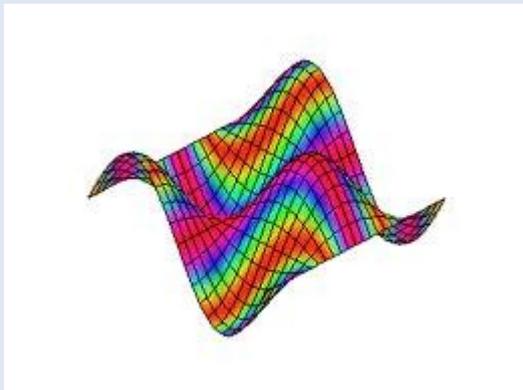
Visualización científica: clasificación de problemas

- Visualización de superficies (2D)
- Visualización de superficies 3D



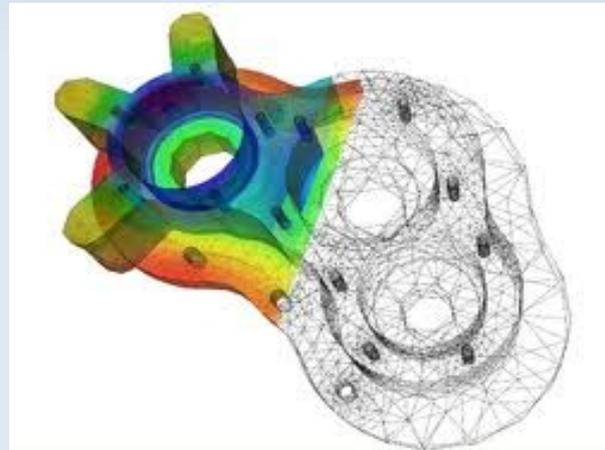
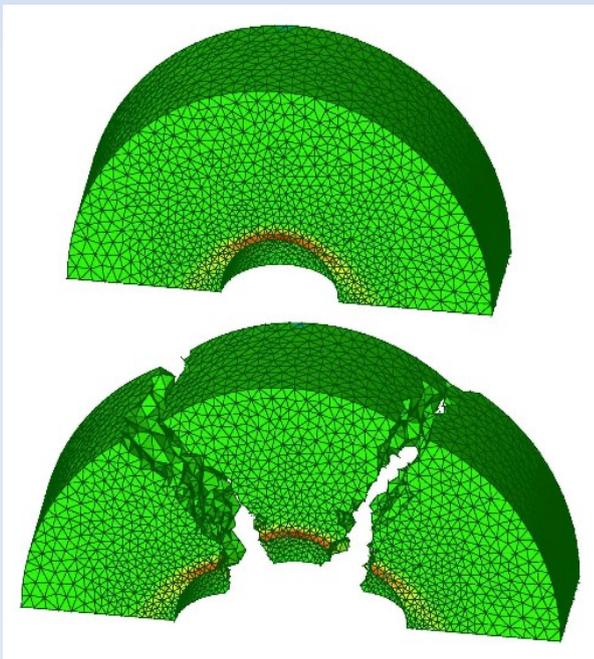
abiertas

cerradas



Visualización científica: clasificación de problemas

- Visualización de datos de Volumen: ejemplo temperatura en un objeto 3D

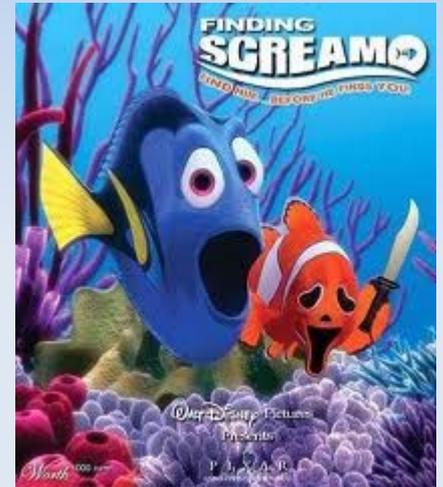


Preguntas fundamentales

- Cómo modelamos superficies en 3D?
- Cómo modelamos objetos con volumen?
- Cómo modelamos funciones matemáticas explícitas?
- Cómo modelamos aproximaciones/discretizaciones de funciones desconocidas? o conocidas?
- Cómo visualizamos estos modelos?

Visualización realista

- Modelamiento matemático computacional de escenas 2D o 3D (computación gráfica 3D)
- No busca rigurosidad sino impresionar
- Aplicaciones:
 - Entretenimiento (juegos, películas)
 - Arte



Usando fractales



Resumen

- Visualización científica y visualización realista son técnicas que interactúan
 - Visualización científica usa elementos de visualización realista
 - Visualización realista usa elementos de visualización científica
-

Computación gráfica 3D

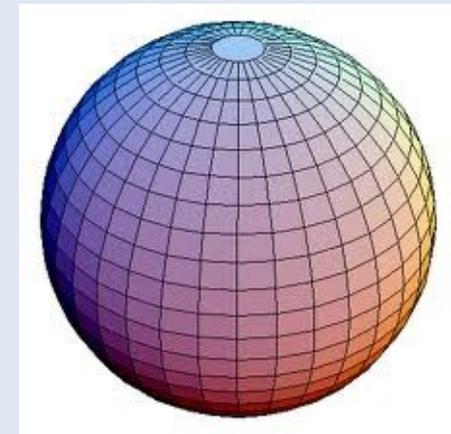
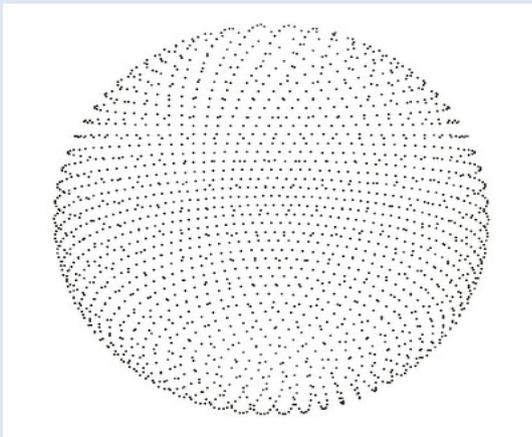
- Modelación matemático computacional de una escena 3D que se visualiza en pantalla/dispositivo 2D
 - Modelación de la escena y sus objetos 3D
 - Transformaciones geométricas
 - Transformaciones de proyección
 - Modelo de eliminación que permite pintar en el dispositivo raster (pantalla discretizada)
 - Algoritmos para: eliminar partes de objetos que no se ven, pintar sobre un dispositivo raster, etc.

Discretización: un concepto muy útil y amplio

- Discretización:
 - Modelo matemático que aproxima y representa un un objeto continuo en base a un número finito de elementos básicos (puntos, polígonos, pedacitos curvos)
 - Representación/modelo computacional asociado al modelo matemático

Distintas alternativas

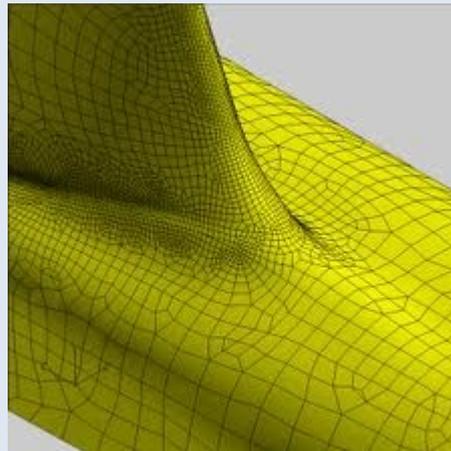
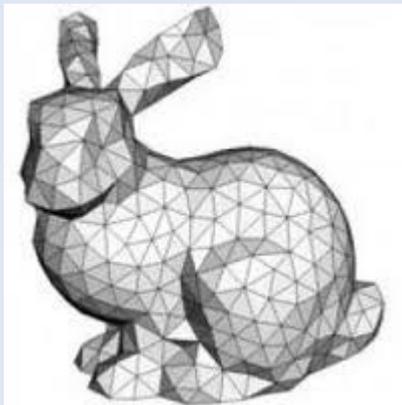
Cómo elegir el mejor? (depende de los requerimientos)



Cómo modelamos/discretizamos objetos 3D?

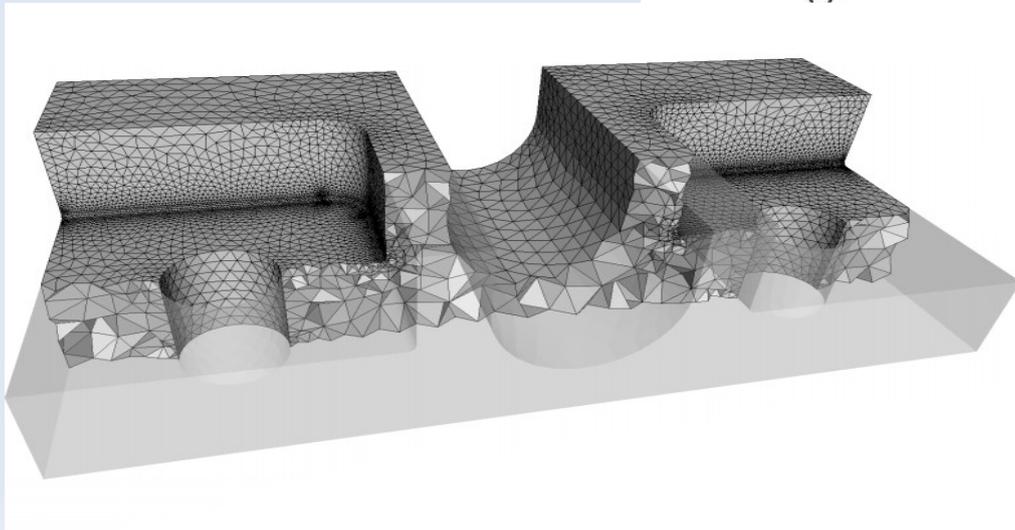
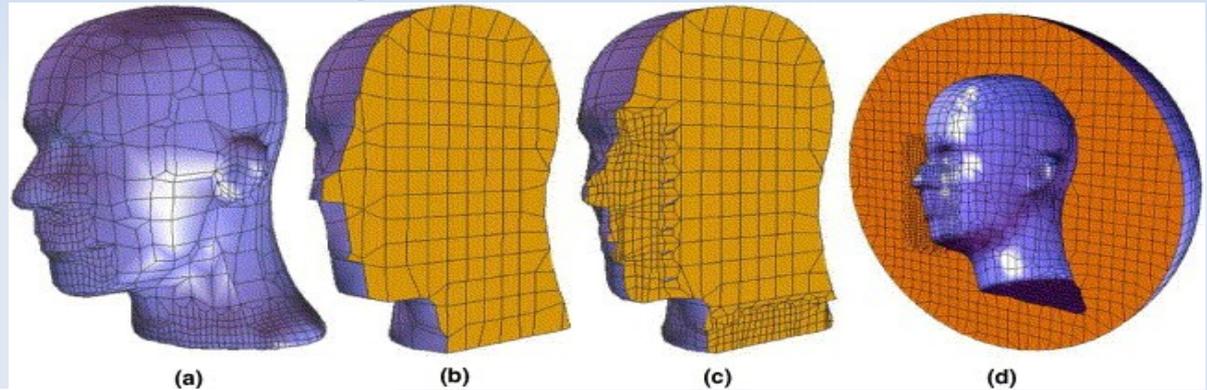
Depende de los requerimientos de la aplicación

- Superficie: triángulos, cuadriláteros (ambos)
 - Más estricta en aplicaciones científicas: restricciones de ángulos (mínimo y máximo), área, etc
 - Más relajada en entretenimientos (juegos, cine): basta que se vea bien



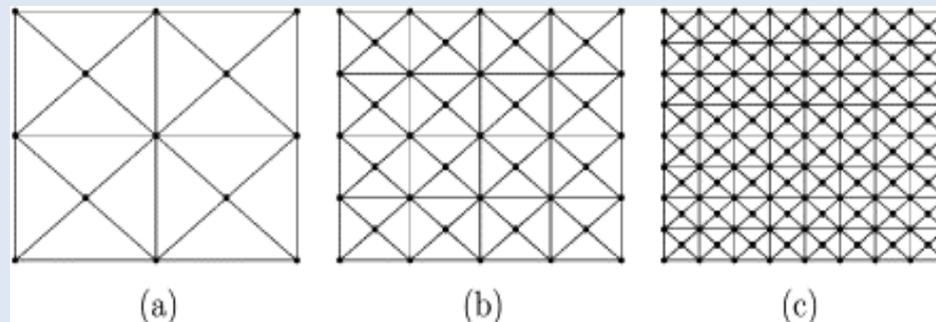
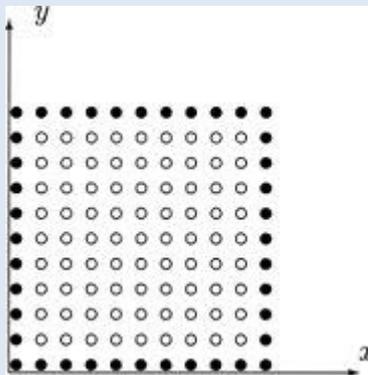
Cómo modelamos/discretizamos objetos 3D?

- Objeto en tres dimensiones (sólido): **tetraedros**, **hexaedros**, pirámides, prismas (todos)
 - Interesa el volumen (interior de los objetos)
 - En aplicaciones científicas: temperatura, fluidos, etc



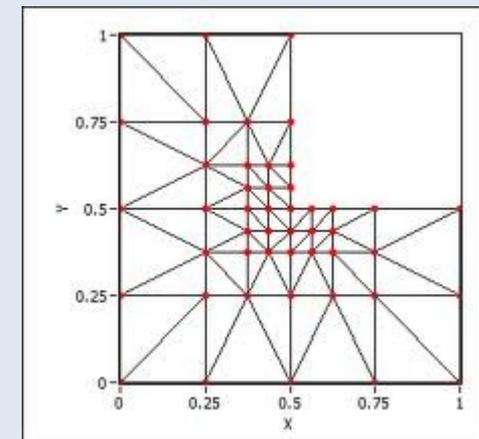
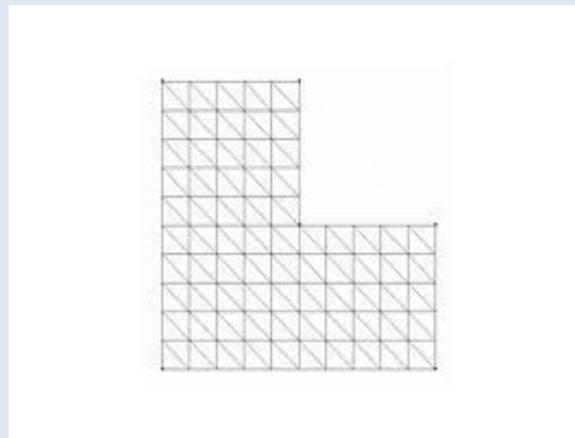
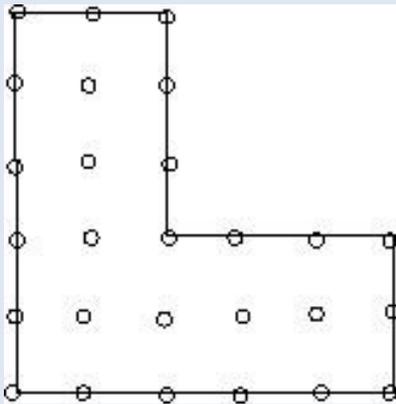
Ejemplos de discretizaciones

- Discretizar una placa cuadrada (geometría simple)
 - **Puntos equidistribuidos:** se le asocia una grilla y se estructura la información en una matriz. **Datos:** conjunto de puntos; **aproximación no continua.**
 - **Malla uniforme de polígonos(triángulos o cuadriláteros):** permite manejar además información de una superficie (normales, área, etc). **Aproximación continua**



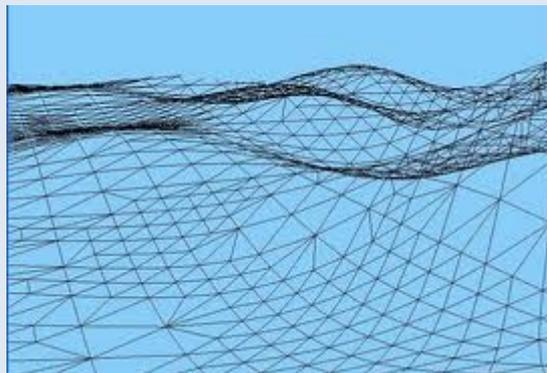
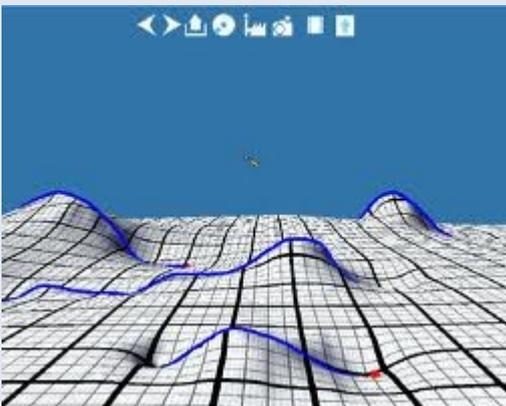
Ejemplos de Discretizaciones

- Discretizar placa 2D con forma de L
 - **Puntos Equidistribuidos:** Se puede asociar una grilla. Matriz? Si, con cuidado.
 - **Malla de polígonos:** permite obtener información de superficie (normales, área, etc)



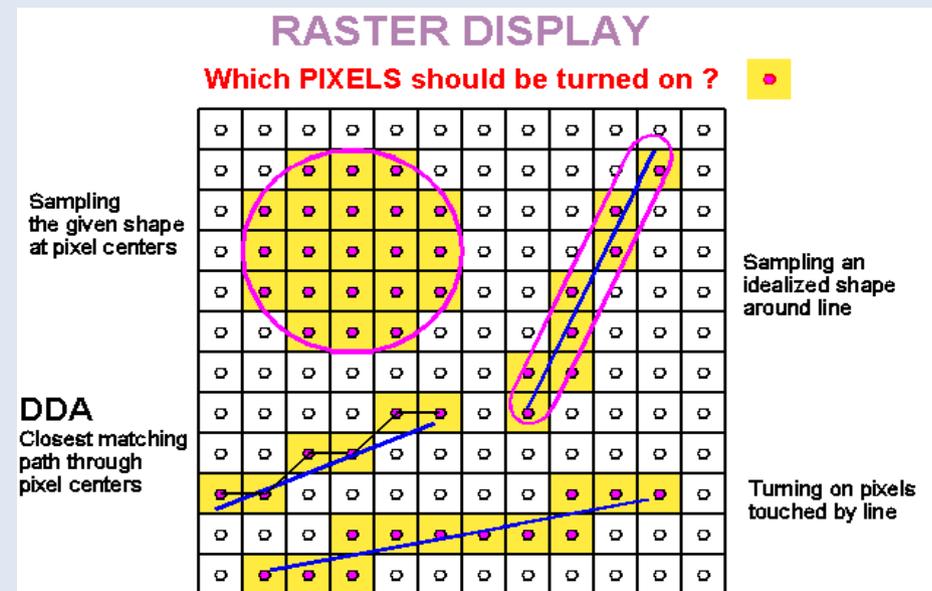
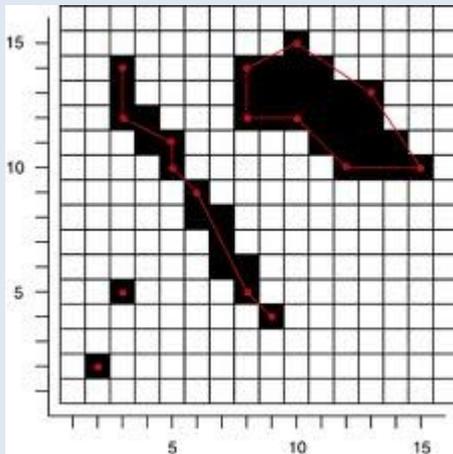
Ejemplos de Discretizaciones

- Discretizar superficie de un terreno. Input:
 - Conjuntos de puntos equidistribuidos (satélite): **posición, altura**. Se asocia grilla y estructura de tipo matriz
 - Cómo visualizar la topografía del terreno?
 - Malla de polígonos: cuadriláteros y triángulos

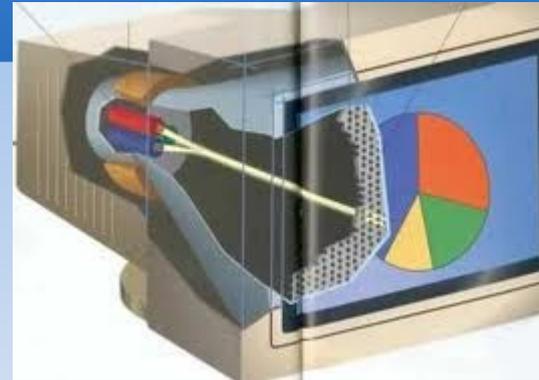


Discretización en hardware gráfico

- Pantalla raster: El modelo es un arreglo rectangular de pixeles
 - Pixel (picture element): Elemento más pequeño (con área distinta de cero) accesible para pintar en el dispositivo raster
 - Resolución: Número total de puntos en cada dirección. Ejes alineados con la pantalla. Normalmente se especifica ancho x alto. Ejemplo 1920x1080.



Discretización: Pantalla raster



- Tecnologías de hardware
 - Tubo de rayos catódicos (CRT)
 - Pantallas de cristal líquido

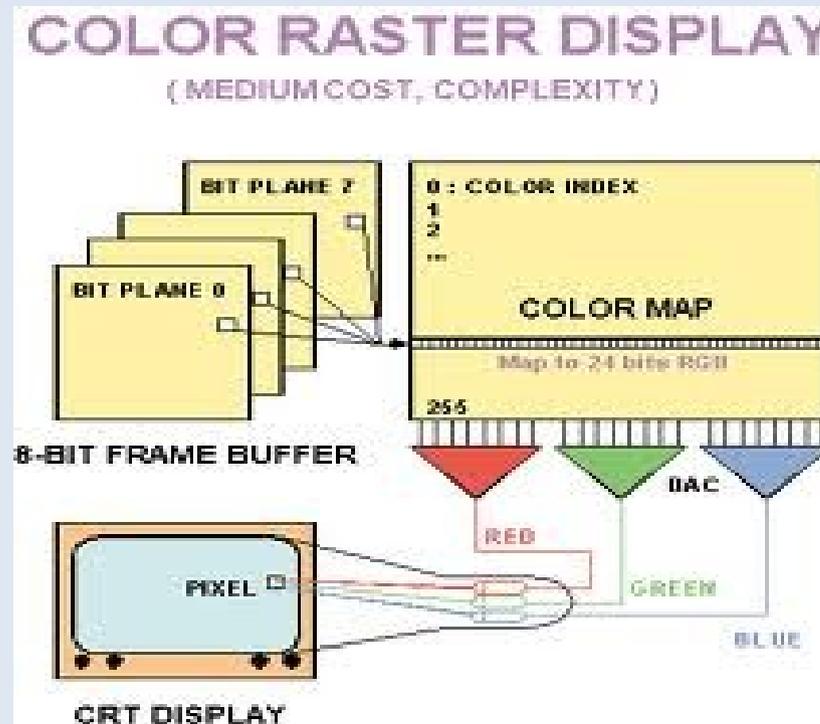
- CRT

- Cada pixel se enciende/pinta por corto tiempo
- Haz de electrones golpea al pixel (fósforo) y se enciende
- Tasa de refresco: Tiempo necesario para encender la imagen nuevamente
- Refresco: **Barrido** según líneas horizontales, de arriba hacia abajo



Frame buffer/ buffer de refresco

- **Frame buffer:** área de memoria (arreglo) donde se almacena la imagen (e información de ésta) asociada a la pantalla completa
 - **Imagen blanco y negro:** 1 bit por pixel
 - **Imagen a color RGB** (red-green-blue): 24 bits/pixel, por ejemplo.



Hardware gráfico actual

- Procesadores:
 - CPU (uno más)
 - Procesador gráfico GPU (Graphic process unit). Acelera proceso gráfico realizando muchas operaciones por hardware y en paralelo

Discretización y métodos numéricos: solución numérica de una ecuación diferencial ordinaria (EDO)

- $y' = dy/dt = f(t,y)$ con $y(0) = 0$ (problema de valor inicial)
- Problema: Encontrar $y(t)$ tal que y pase por $(t_0, y_0) = (0,0)$
 - No siempre existe una solución explícita (no es posible hay una fórmula que represente la solución de $y(t)$) entonces podemos encontrar una solución aproximada
- Método de Euler
 - Encontraremos un número finito de puntos (t_k, y_k)
 - Cómo encontrar estos puntos en el intervalo $[a,b]$ ($a = 0$)
 - $t_k = a + kh$ para $k = 0, 1, \dots, m$ donde $h = (b-a)/m = b/m$
 - $t_{k+1} = t_k + h$

Discretización y métodos numéricos

- Suponiendo que $y(t)$, $y'(t)$, $y''(t)$ son continuas y usando el teorema de Taylor para desarrollar $y(t)$ alrededor de $t=0$, para cada punto t existe un punto c_1 entre t_0 y t tal que
 - $y(t) = y(t_0) + y'(t_0)(t-t_0) + y''(c_1)(t-t_0)^2 / 2$
- Al sustituir $y'(t_0) = f(t_0, y(t_0))$ y $h = t_1 - t_0$
 - $y(t_1) = y(t_0) + f(t_0, y(t_0))h + y''(c_1)h^2 / 2$
- Si h es suficientemente pequeño podemos despreciar el término que contiene h^2 y obtener
 - $y(t_1) = y(t_0) + f(t_0, y(t_0))h$ (aproximación de Euler)
- Repitiendo el proceso podemos formular el paso general
 - $y(t_{k+1}) = y(t_k) + f(t_k, y(t_k))h$

Ecuaciones diferenciales parciales (EDP) elípticas

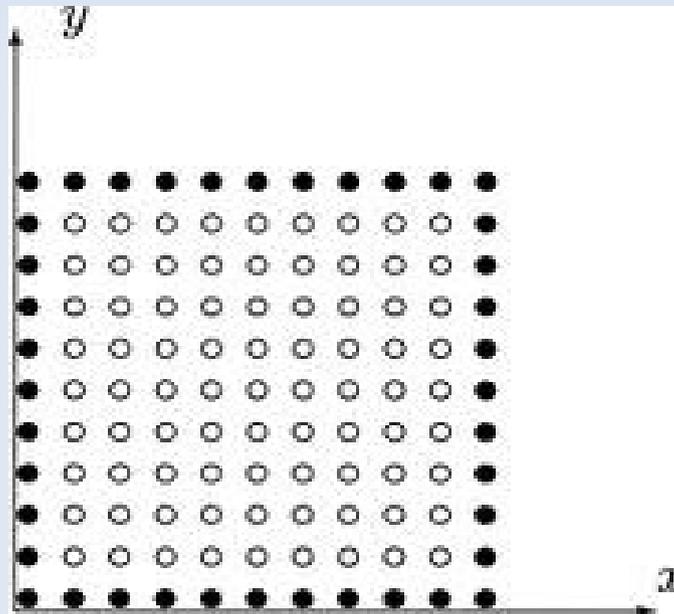
- Muchos problemas en física e ingeniería se modelan matemáticamente con ecuaciones diferenciales parciales
- Una ecuación diferencial en donde aparecen dos o más variables independiente se llama **ecuación en derivadas parciales**
- Ejemplo: **Ecuación de Laplace**
 - $\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = 0$ en Ω
 - Condiciones de borde
 - Dirichlet $u = f_1(x, y)$
 - Neumann $\frac{\partial u}{\partial n} = f_2(x, y)$
- Modela fenómenos estacionarios: potencial electrostático o distribución de temperatura

Métodos numéricos para EDPs

- Diferencias finitas
- Elementos finitos
- Volúmenes finitos

Métodos de diferencias finitas para ecuación de Laplace 2D

- Dominio de aproxima por una grilla



Diferencias Finitas

- Simple e intuitivo
- Basado en aproximar operadores diferenciales mediante operadores de diferencias

$$\nabla^2 u_{i,j} \approx \frac{u_{i+1,j} + u_{i-1,j} + u_{i,j+1} + u_{i,j-1} - 4u_{i,j}}{h^2}$$

- Error $O(h^2)$
- u_{ij} aproxima el valor de $u(x_i, y_j)$

Solución numérica

- **Solución numérica:** es la aproximación de la solución exacta en el conjunto de puntos del dominio
 - Se escribe una ecuación de diferencias por cada punto de la grilla
 - Se agregan ecuaciones asociadas a las condiciones de borde
 - Se obtiene un sistema lineal de ecuaciones $A\tilde{u}=b$ en las variables u_{ij}
 - Se resuelve por métodos iterativos o directos
- La solución exacta de la EDP es una superficie definida sobre el dominio
- La solución numérica es una función discreta definida sobre el dominio

Herramienta a usar para obtener soluciones numéricas

- **Matlab**
 - software matemático
 - lenguaje técnico de alto nivel procedural, basado en matrices
 - aplicaciones numéricas, análisis de datos, visualización de datos
- Empresa: mathworks.com .
- Sitio de matlab: (demo y tutoriales)

<http://www.mathworks.com/products/matlab/>