

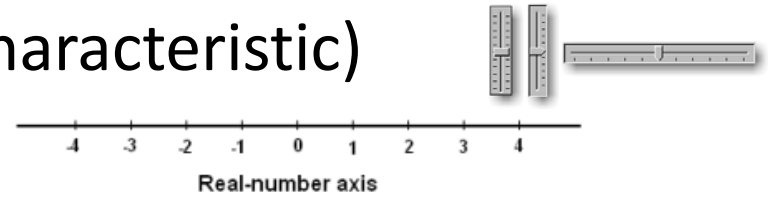
Data and Its Representation

Goal: know the common data forms that visualization is dealing with; how to classify them and how to store them. This is important for correctly loading the data for visualization.

Data – Dimensions

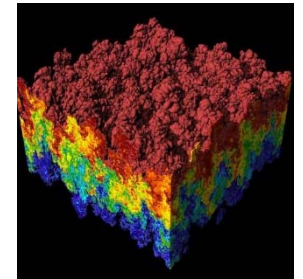
- **Data Dimensionality** (Intrinsic Characteristic)

- **1D**: slider bars for scalar value range

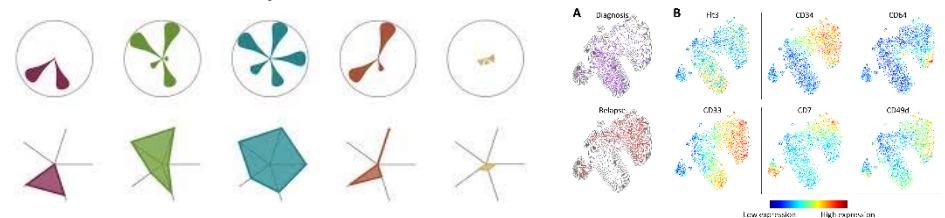


- **2D**: geographical data, images, maps, 2D slices of 3D data...

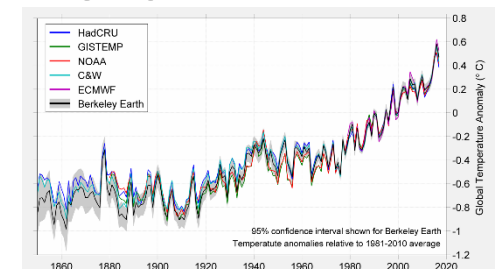
- **3D**: CAD, architecture, medical, biological, scientific computing, ...



- **N-D**: records in logs, data entries in database, social media ...



- **Time-series**: things that are changing over time for any dimensional data!!



Data – Storage

- **Data organization (representation)**
 - **Sequential:** lists (linear relation)
 - **Rational:** tables (databases)
 - **Tree:** hierarchical data, nested data
 - **Graphs:** various networks, relations

```
ply
format ascii 1.0
comment created by platoply.
element vertex 8
property float32 x
property float32 y
property float32 z
element face 6
property list uint8 int32 vertex_indices
end_header
-1 -1 -1
1 -1 -1
1 1 -1
-1 1 -1
-1 -1 1
1 -1 1
1 1 1
-1 1 1
4 0 1 2 3
4 5 4 7 6
4 6 2 1 5
4 3 7 4 0
4 7 3 2 6
4 5 1 0 4
```

header

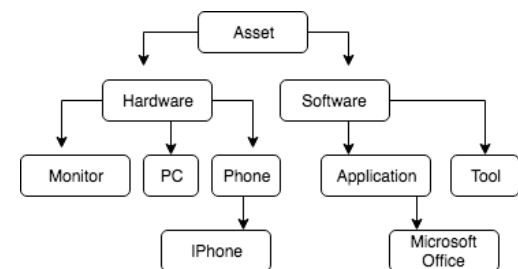
Vertex list

Face list

Data

When **stored**, the data is either stored linearly (i.e., one data point after another) or stored in databases.

Salesperson	Country	2011		2012		2013		Total	
		Units	Order Amt	Units	Order Amt	Units	Order Amt	Units	Order Amt
Bromley	UK	232	24,756.89	228	40,396.64	73	9,894.51	533	75,048.04
Bromley	USA	58	7,553.95	27	3,654.00	7	1,101.20	92	12,309.15
Callahan	USA	623	49,400.07	337	43,263.95	200	18,059.50	1,160	110,723.52
Coghill	UK	81	4,029.25	39	4,657.11			120	8,686.36
Coghill	USA	885	120,626.31	520	46,505.90	405	49,945.11	1,810	217,077.32
Farnham	UK	170	14,055.87	44	5,892.65	17	2,560.40	231	22,508.92
Farnham	USA	699	89,663.20	506	73,360.59	217	15,663.56	1,422	178,687.35
Finchley	USA	699	95,850.36	487	55,787.97	302	30,861.76	1,488	182,500.09
Fulter	USA	539	71,168.14	473	73,524.18	170	17,811.46	1,182	162,503.78
Gillingham	UK	397	40,826.37	276	17,181.58	202	14,519.68	875	72,527.63
Gloucester	UK	209	31,433.16	143	19,691.89	135	17,667.20	487	68,792.25
Rayleigh	UK	422	59,827.19	268	41,903.64	131	15,232.16	821	116,962.99
Grand Total		5,014	609,190.76	3,348	425,820.10	1,859	193,316.54	10,221	1,228,327.40



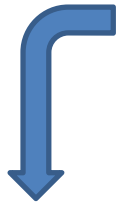
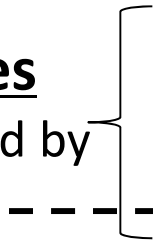
LET US LOOK AT SCIENTIFIC DATA

How to Classify Scientific Data?

- Characteristics of datasets:
 - dimension of domain: number of coordinates or (independent) parameters
 - **dimension of values (number of dependent values in a data point)**
 - static vs. time-dependent

Data Types

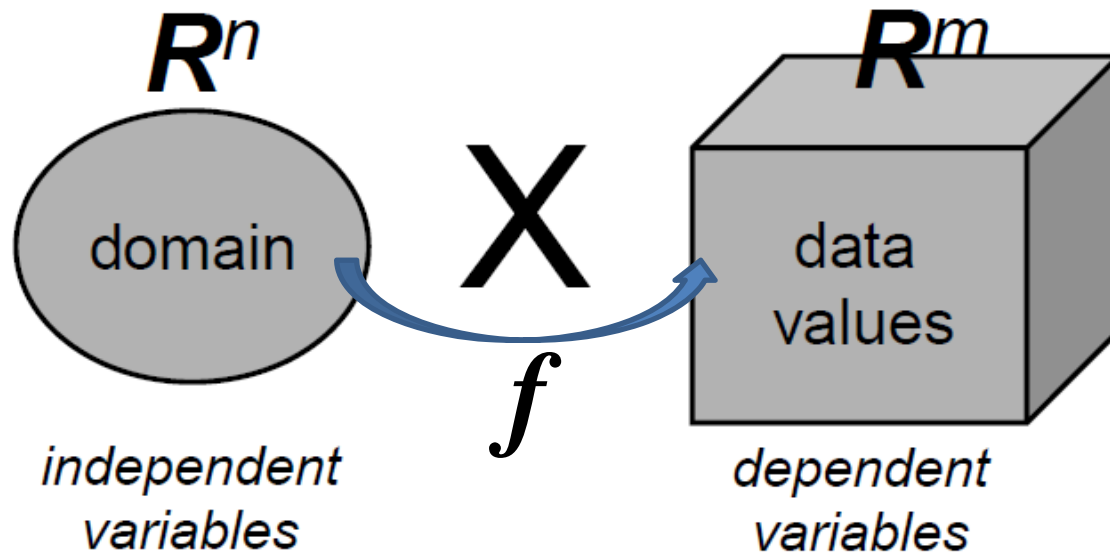
determined by



- discretized data
 - type of discretization: (un-)structured grid, scattered data, ...
- deterministic vs. stochastic (uncertain)

Representation

In many cases, scientific data describe certain functions with the input as the spatial coordinates and time (**domain**), and the output as the **data values**.



scientific data

$$\subseteq R^{n+m}$$

Source: VIS, University of Stuttgart

n

m

1D

scalar

2D

vector

3D

tensor

+time

A summary of Data Types

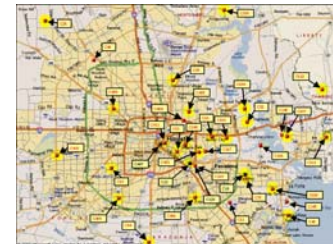
Scientific Data Representation

- In the *scientific data* regime, the data typically describe continuous physical events in the continuous physical space+time.
- This contradicts the discrete nature of the digital machine.
- In order to represent these data in the machine with finite space, **discrete representation** is necessary
 - how to sample (a question for data collection)
 - Scattered points, no grid
 - structured vs. unstructured (polygonal meshes)
 - Sampling case — Cartesian grids, images, random, etc.
 - Unstructured, connectivity, spacing, primitives (e.g., triangles, tet, quads...)
- Continuous — basis, e.g., polynomials, spectral, wavelet

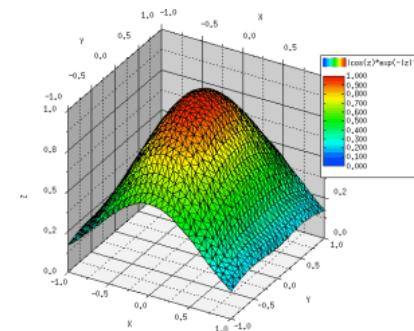
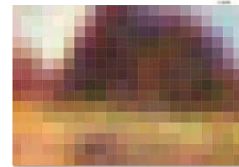
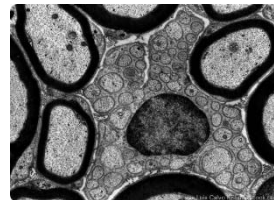
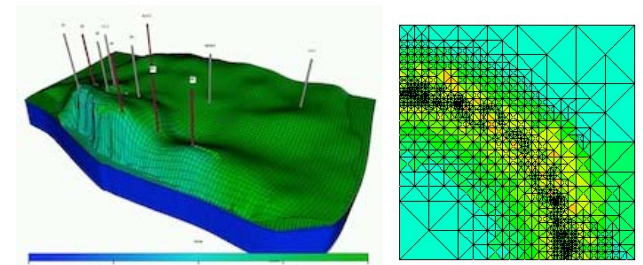
Domain/Space Discretizations

Most time, the discretization strategy is determined by the types of data sources :

- Measurement data:
 - typically scattered (no grid/mesh)
- **Numerical simulation data:**
 - structured, block-structured, unstructured grids
 - adaptively refined meshes
 - etc.
- Imaging methods:
 - uniform grids
- Mathematical functions:
 - uniform/adaptive sampling on demand

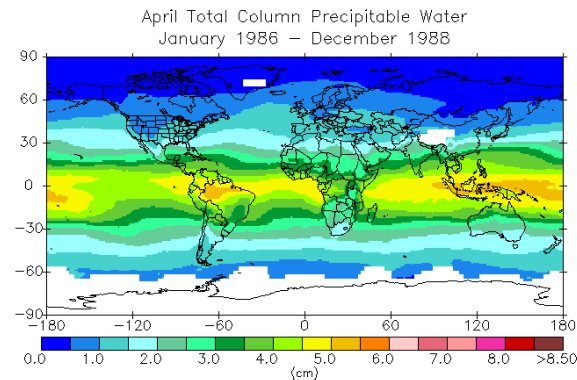
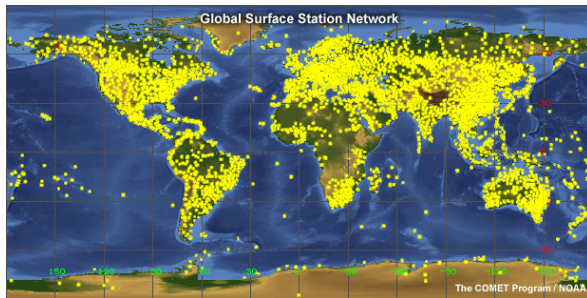


Houston air monitor stations



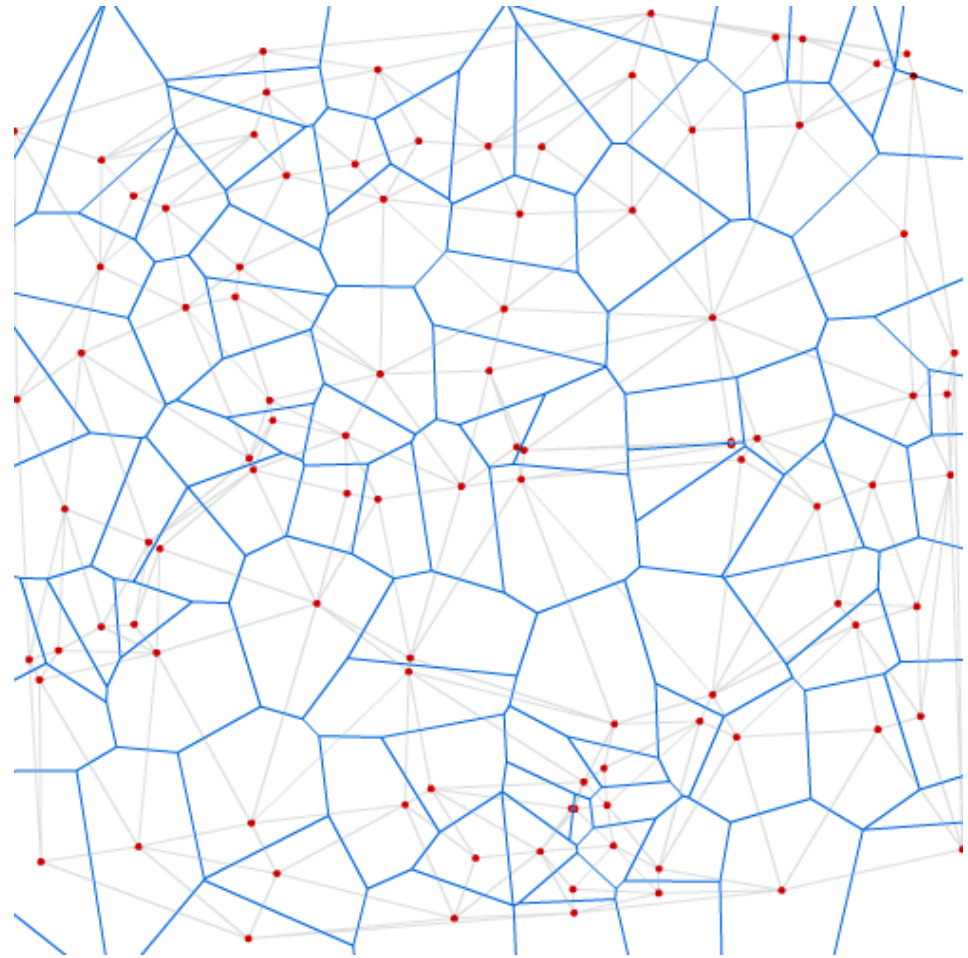
Scattered Data

- Scattered data means: only nodes, no cells
- Typical data sources: measurement data, e.g., meteorological

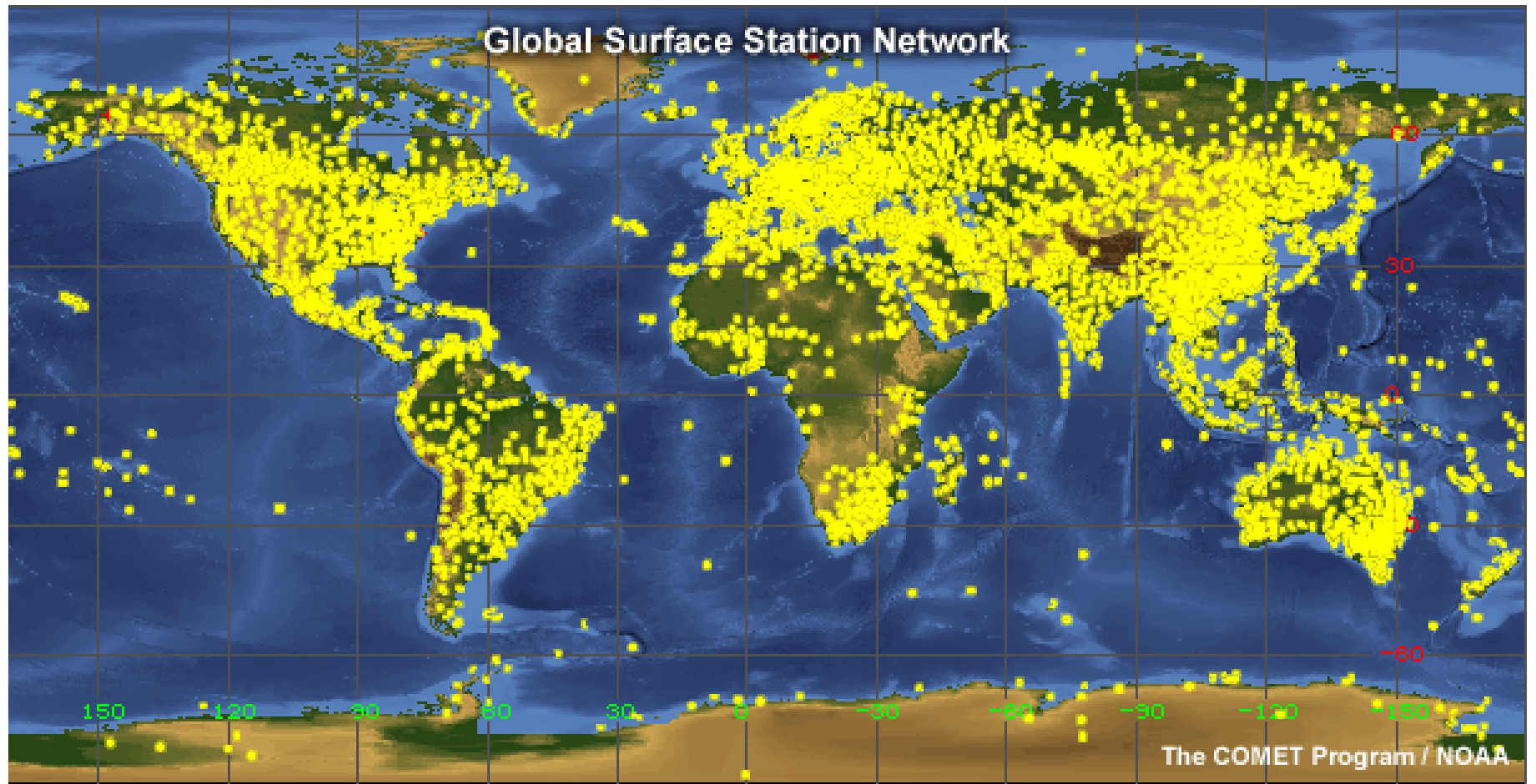


- Options for visualization:
 - point-based methods (relatively few algorithms)
 - triangulation (next slide), e.g., constrained Delaunay, difficult in 3D
 - resampling (two slides later) on uniform grid

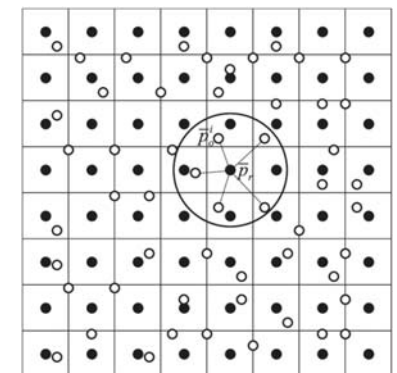
Delaunay Triangulation



Source: <http://en.nicoptere.net/?p=10>



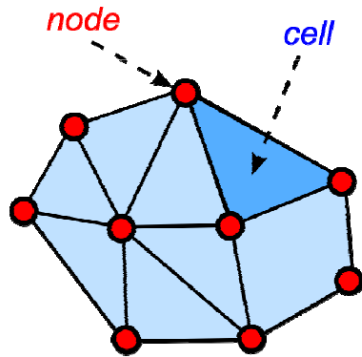
Scattering points to regular grid re-sampling



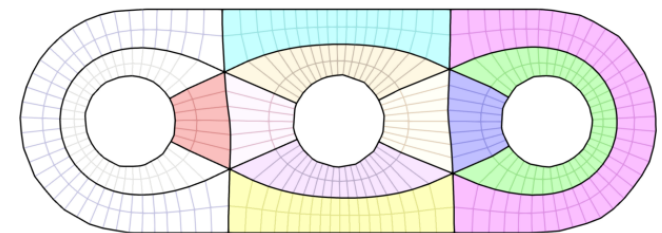
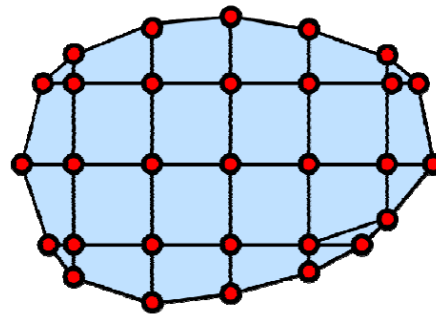
**LET US LOOK AT DATA STORED ON
GRIDS (WITH 2D OR 3D CELLS)**

Data Stored on Grids – Unstructured

- Typical data sources: simulation data, e.g., CFD
- **2D** (plane or surfaces) unstructured
 - cells are triangles and/or quadrangles
 - domain can be a surface embedded in 3-space



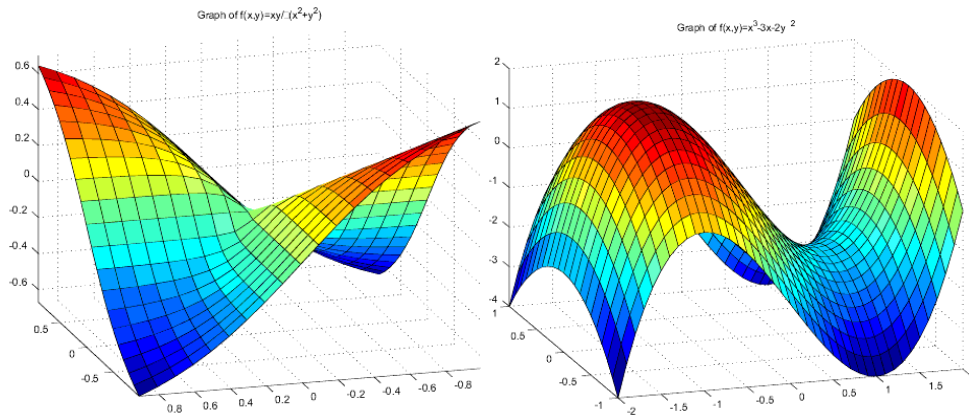
unstructured grids



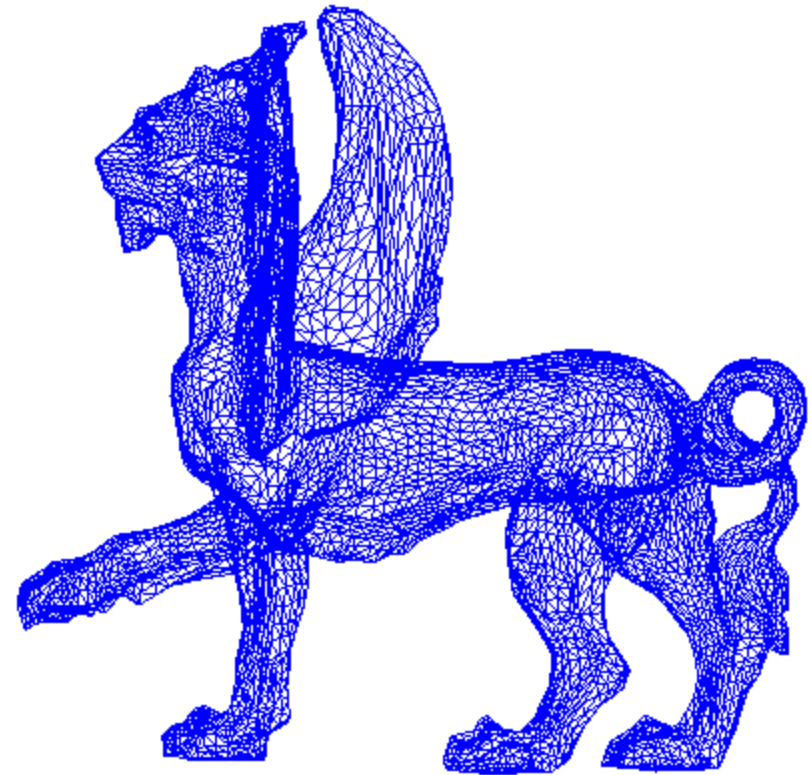
semi-structured

Arbitrary Surfaces

- **Mesh (geometry)**
 - Discrete representation



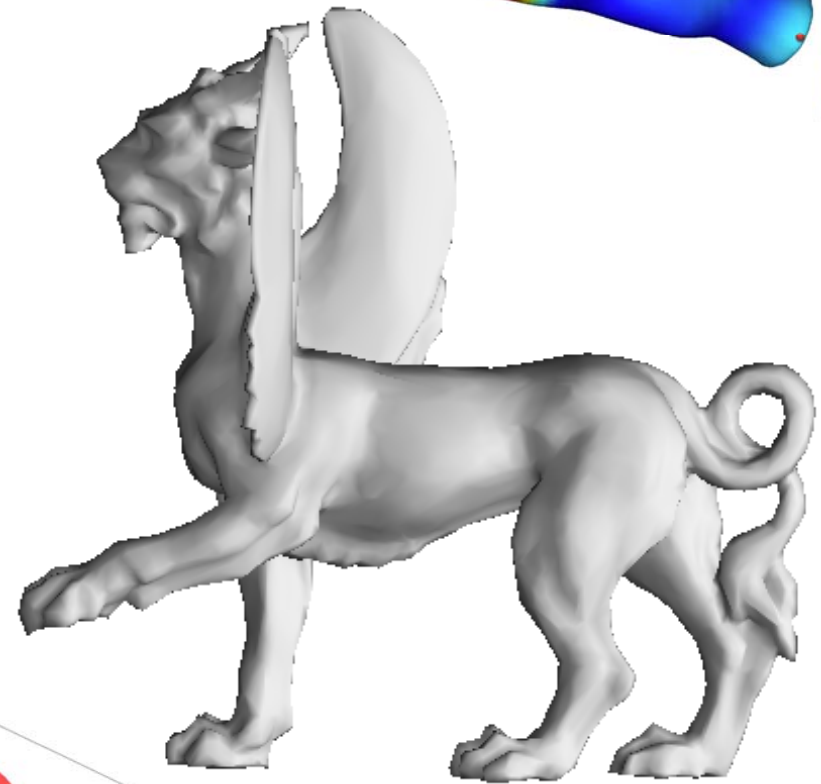
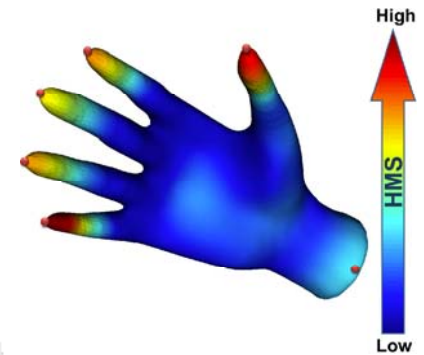
Parametric surfaces



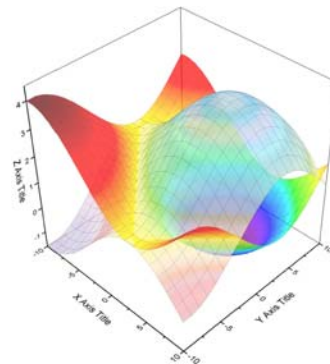
Triangular mesh

Arbitrary Surfaces

- Shape visualization
- **Attributes** that can be used by visualization:
 - Shading/lighting
 - Silhouette
 - Feature curves
 - **Colors**
 - **Transparency**



Surface with shading



What do we need to represent an unstructured grid in its processing?

- For 2-manifold surfaces:

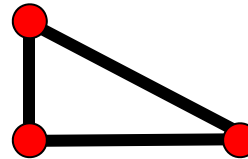
- Vertex (0D)



- Edge (1D)



- Face (2D)



- Polyhedron (contains the above member variables)

What need to be stored in the data file?

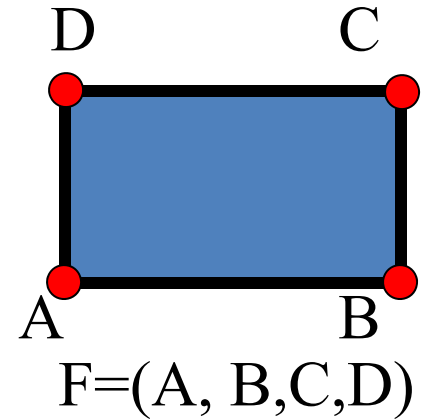
Vertex (required)

● $V=(x, y, z)$

- Basic
 - (x, y, z) – coordinates, necessary
 - index – almost always needed, automatic based on ordering
 - Attributes or data, like (n_x, n_y, n_z) – normal, optional
- Derived
 - List of faces incident to the vertex - almost always needed, constructed later
 - List of edges incident to the vertex – almost always needed, constructed later

What need to be stored in the data file?

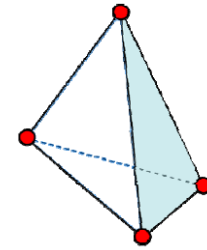
Face (required)



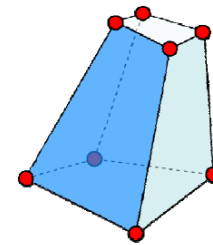
- Basic
 - List of vertices - necessary, typically the indices of the vertices
 - index – almost always needed, automatic based on ordering
 - Attributes and data, optional
- ↓
- *Derived*
 - *List of edges – almost always necessary, constructed later*

What Should be Stored for 3D Grids?

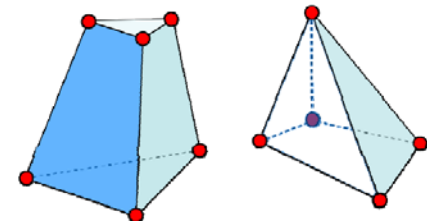
- Tetrahedral
 - Vertex list
 - 3D cell list with four vertices for each cell



- Hexahedra
 - Vertex list
 - 3D cell list with edges (pairs of vertices) that form a cell

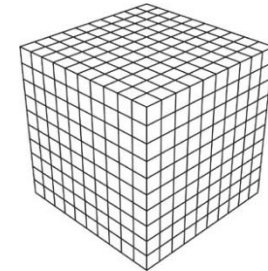
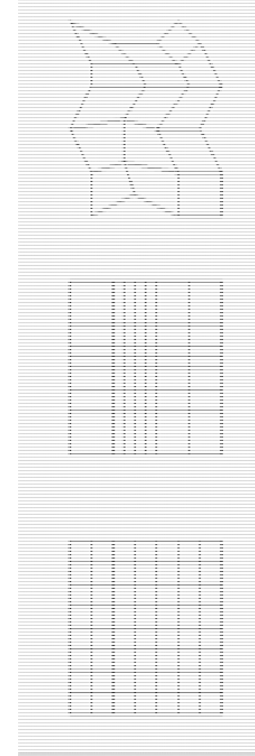


- Hybrid - Polyhedra
 - Vertex list
 - 2D cell list
 - 3D cell list



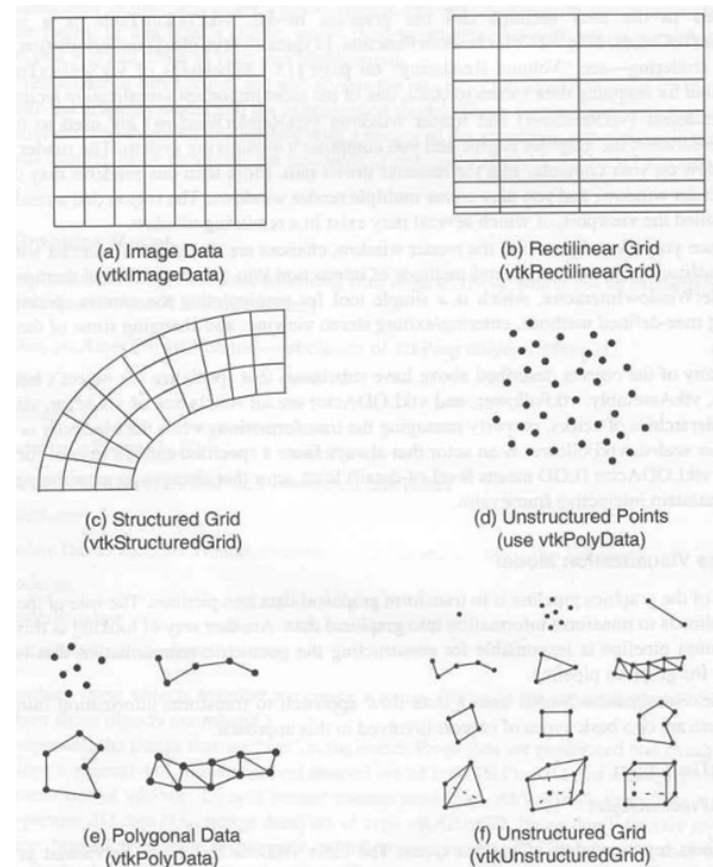
Structured Grids – Usually Axis-Aligned

- General case: *curvilinear* grid
 - nodes given in array $N_i \times N_j \times N_k$
 - cells are implicit
- Special case: *rectilinear* grid
 - simpler coordinate functions:
 $x = x(i), y = y(j), z = z(k)$
- More special: *uniform* grid
 - coordinates defined by axis-aligned bounding box (2 points)



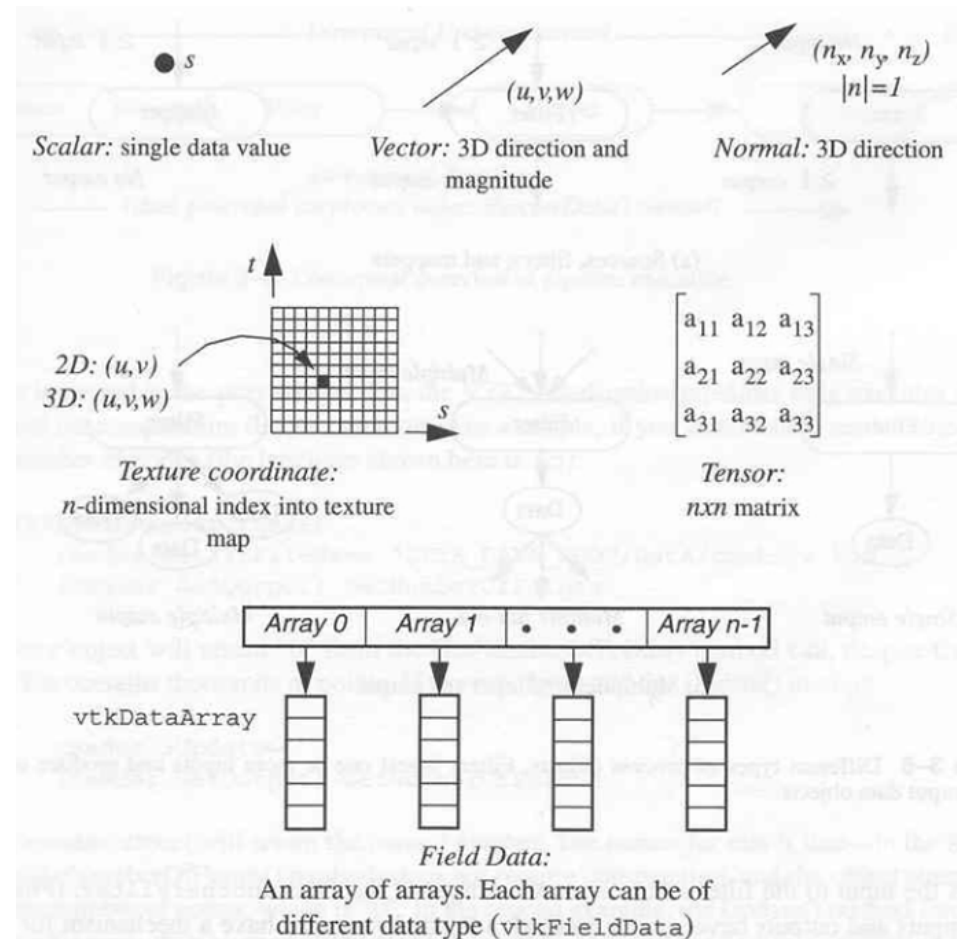
VTK Data Format

- Dataset types found in VTK
 - Image data
 - Rectilinear grid
 - Structured grid
 - Unstructured grid
 - Unstructured points
 - Polygonal data
- Data objects have geometric and topological structure (points and cells)
- Cells are topological arrangements of points



VTK Data Format

- Types of data values associated with points and/or cells
 - scalar
 - vector
 - normal
 - texture coordinate
 - tensor
 - field data
 -



Acknowledgment

- Thanks for materials from
- Dr. Eugene Zhang
- Dr. Ronald Peikert
- Dr. Carlos A. Vinhais