Data and Its Representation
General Data Types I

- **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
General Data Types II

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

When *stored*, the data is either stored in sequential or rational (tables)
LET US LOOK AT SCIENTIFIC DATA
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)
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.

A summary of Data Types

<table>
<thead>
<tr>
<th>$n$</th>
<th>$m$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1D</td>
<td>scalar</td>
</tr>
<tr>
<td>2D</td>
<td>vector</td>
</tr>
<tr>
<td>3D</td>
<td>tensor</td>
</tr>
<tr>
<td>+time</td>
<td></td>
</tr>
</tbody>
</table>

Source: VIS, University of Stuttgart
Data Representation

• In the *scientific data* regime, the data typically describe continuous physical events in the continuous physical space+time.

• This contradicts the discrete natural of the digital machine.

• In order to represent these data in the machine with finite space, *discrete representation* is necessary
  • how to sample — structured vs. unstructured (polygonal meshes)
    • Sampling case — Cartesian grids, images, spacing, etc.
    • Unstructured, connectivity, spacing, primitives (e.g. simplexes)
    • Lists

• Continuous — basis, e.g. polynomials, spectral, wavelet
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
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, e.g., constrained Delaunay, difficult in 3D
- resampling on uniform grid
Delaunay Triangulation

Source: http://en.nicoptere.net/?p=10
resampling on uniform grid
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
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 the 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)

• Basic
  – (x, y, z) – coordinates, necessary
  – index – almost always needed, automatic based on ordering
  – Attributes or data, like (nx, ny, nz) – 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

\[ V = (x, y, z) \]
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 $Ni \times Nj \times Nk$
  – 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