Web Technologies for Bioinformatics

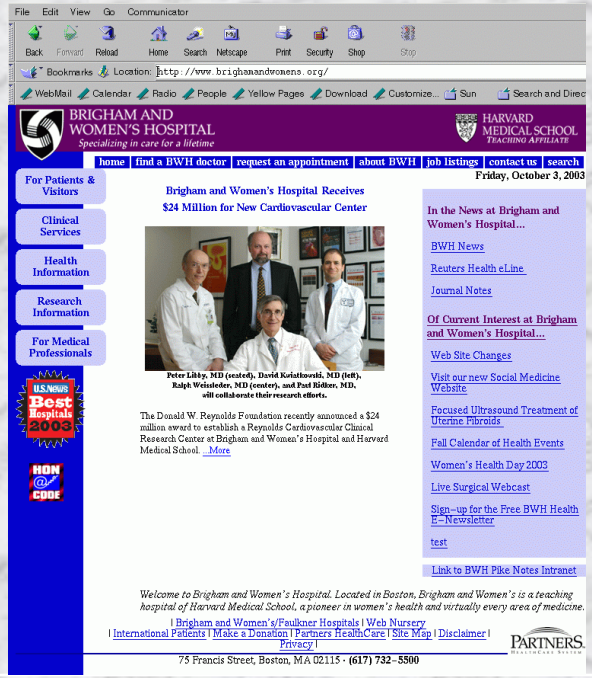
Ken Baclawski

Data Formats

- Flat files
- Spreadsheets
- Relational databases
- Web sites

```
011500   18.66   0   0  62   46.27102
011500   26.93   0   1  63   68.95152
020100   33.95   1   0  65   92.53204
020100   17.38   0   0  67   50.35111
```

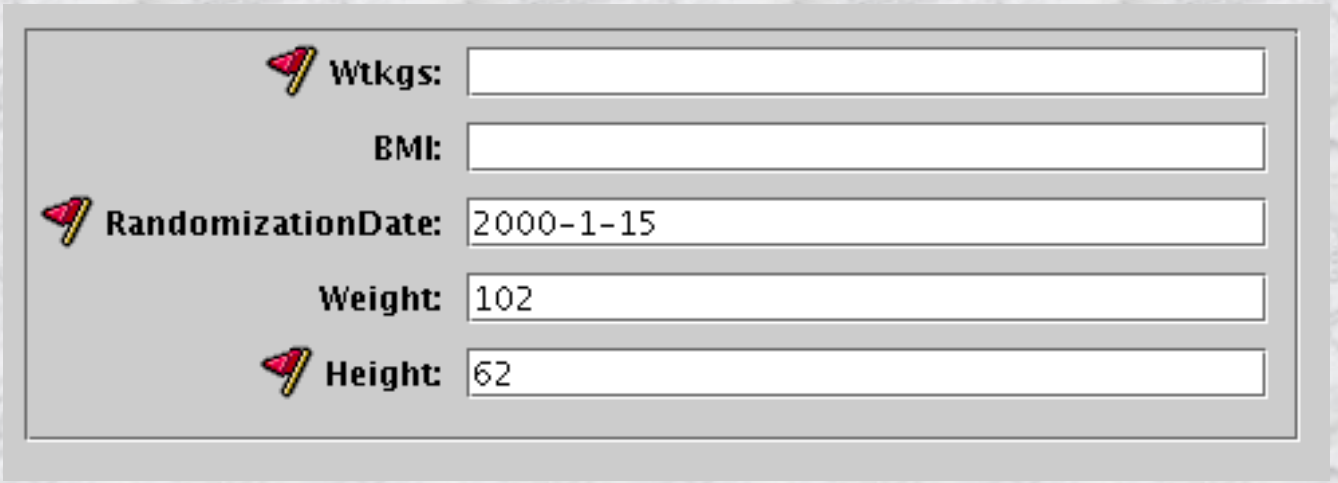
component	variable	initial_value	physical_unit	interface
membrane	u	-85.0	millivolt	out
membrane	Vr	-75.0	millivolt	out
membrane	Cm	0.01	microF_per_mm2	
membrane	time		millisecond	in
ionic_current	I_ion		microA_per_mm2	out
ionic_current	v			in
ionic_current	Vth		millivolt	in



XML Documents

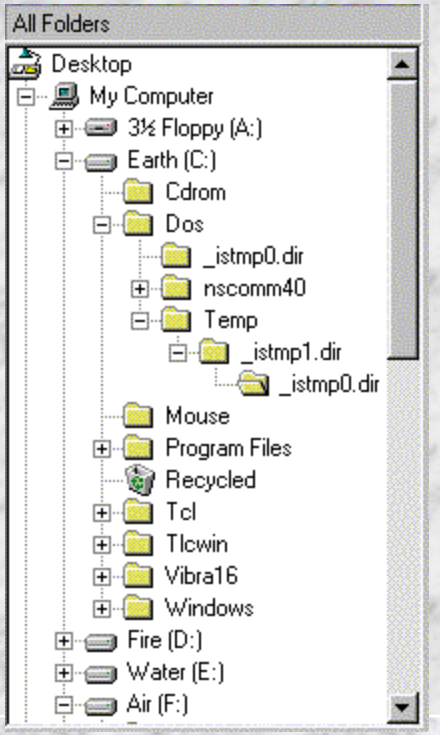
- Flexible very popular text format
- Self-describing records

```
<Interview RandomizationDate="2000-01-15" BMI="18.66" Height="62" Weight="102" ... />
<Interview RandomizationDate="2000-01-15" BMI="26.93" Height="63" Weight="152" ... />
<Interview RandomizationDate="2000-02-01" BMI="33.95" Height="65" Weight="204" ... />
<Interview RandomizationDate="2000-02-01" BMI="17.38" Height="67" Weight="111" ... />
```



XML Documents (continued)

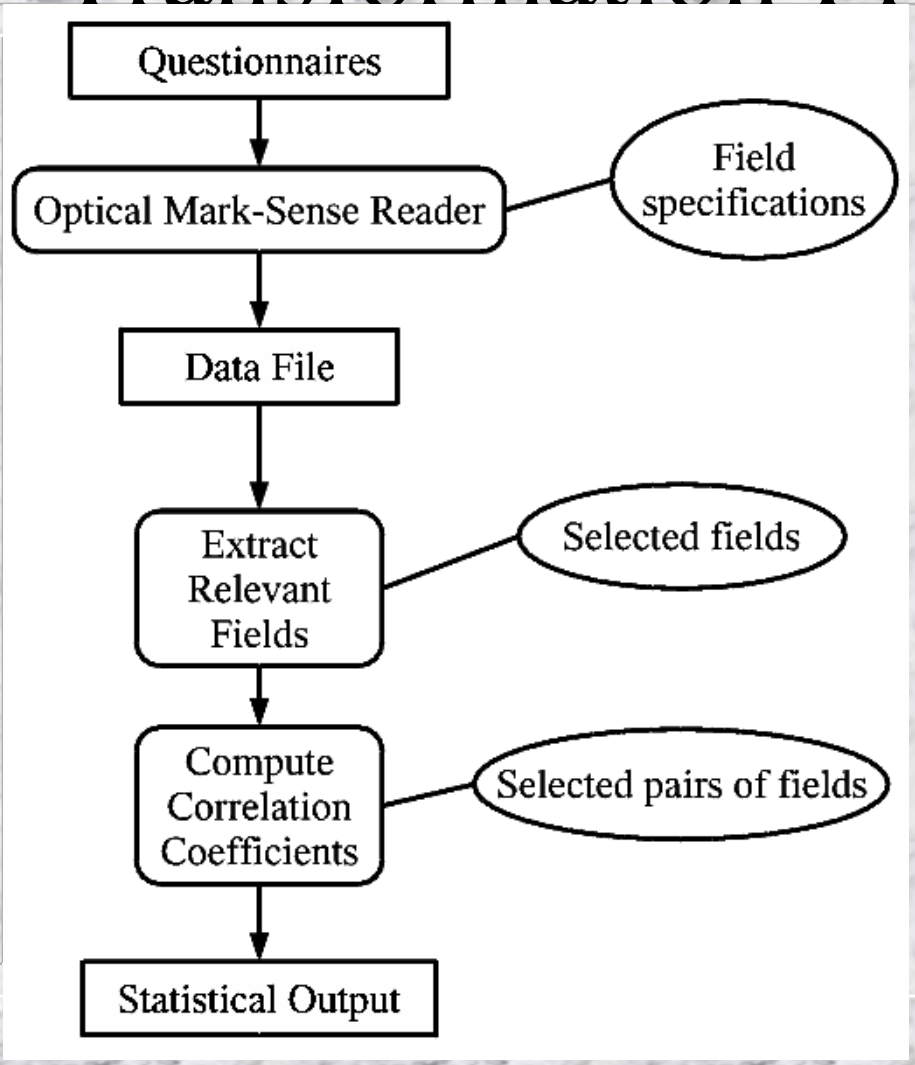
- Hierarchical structure



Purpose of Data

- Data is collected and stored for a purpose.
- The format serves that purpose.
- Using data for another purpose is common.
- Data presentation (such as on a Web site) is one example of such a use.
- It is important to anticipate that data will be used for many purposes.
- Data is reused by *transforming* it.

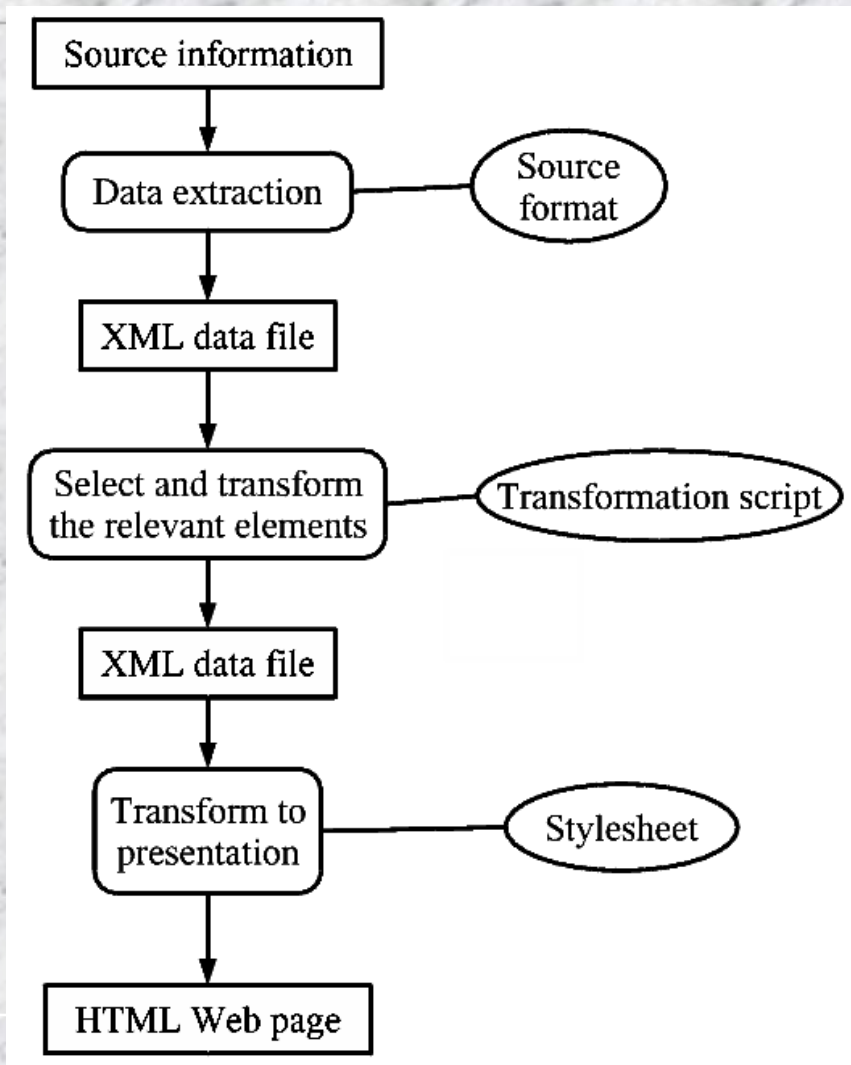
Statistical Analysis as a Transformation Process



- Transformation consists of a series of steps.
- Specialized equipment and software is used for each step.
- Separation into steps reduces the overall effort.

Web Site Construction

- Web sites can be constructed using a Web site authoring tool (e.g., Front Page).
- Alternatively, one could use a transformation process to separate concerns.



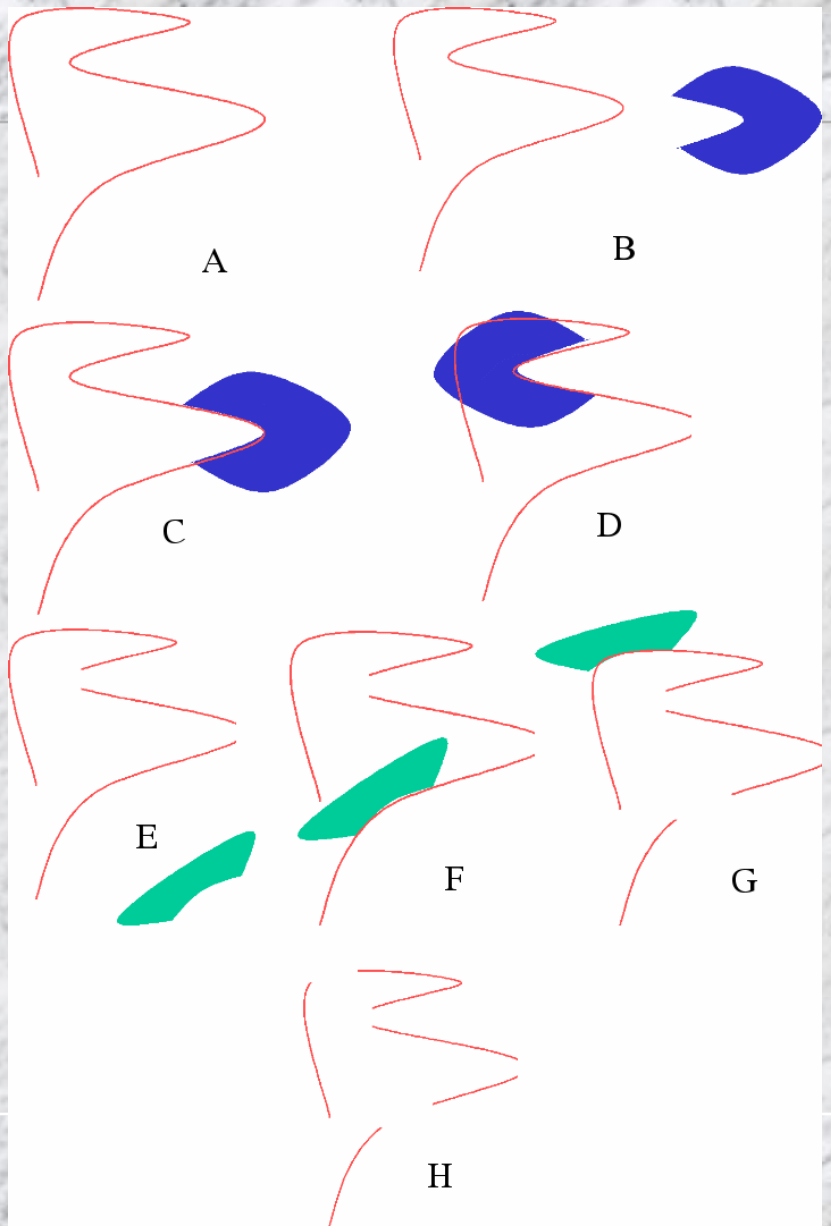
Advantages of Transformation

- Reduces the overall effort.
- Presentation style is independent of the source content.
- Presentation style can be changed with immediate effect.
- Uniform enforcement of presentation style.
- Updates to content are immediate.
- Content can be used for many other purposes:
  - **Many reports in many formats**
  - **Proposals**
  - **Data sharing with other institutions**
  - **Data mining**

Transformation Languages

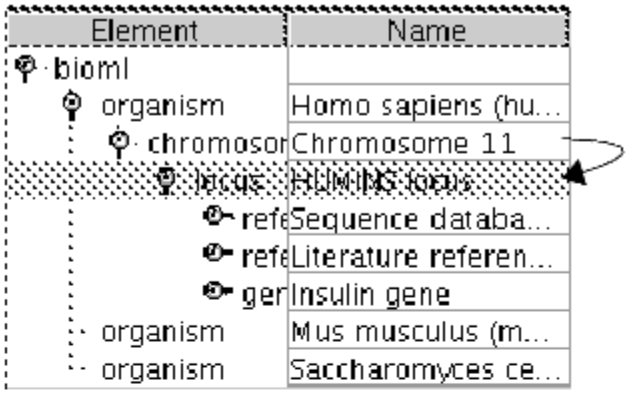
- Traditional programming languages such as Perl, Java, etc.
- Rule-based (declarative) languages such as the XML Transformation language (XSLT).
  - Rule-based rather than procedural
  - Transform each kind of element with a template
  - Matching and processing of elements is analogous to the digestion of polymers with enzymes.

Transformation as Digestion



- The blue enzyme attacks the polymer at two locations.
- The resulting three polymers are then attacked by the green enzyme.

XSLT “Digestion”




```
<xsl:template match="chromosome">
  ...
  <xsl:apply-templates select="locus"/>
</xsl:template>


<xsl:template match="locus">
  ...
</xsl:template>
```

- An XSLT program consists of templates
- Each template processes a set of matching elements
- A template can break up the element to be processed by other templates

```
<?xml version="1.0"?>
<xsl:transform version="1.0”
 xmlns:xsl="http://www.w3.org/1999/XSL/Transform">

<!-- Change all occurrences of P to Protein -->
<xsl:template match="P">
  <Protein>
    <xsl:apply-templates select="@*|node()"/>
  </Protein>
</xsl:template>

<!-- Change all occurrences of S to Substrate -->
<xsl:template match="S">
  <Substrate>
    <xsl:apply-templates select="@*|node()"/>
  </Substrate>
</xsl:template>

<!-- Don't change anything else -->
<xsl:template match="@*|node()">
  <xsl:copy>
    <xsl:apply-templates match="@*|node()"/>
  </xsl:copy>
</xsl:template>

</xsl:transform>
```

```
<Array><P id="Mas375"><interactionsubstrate="Sub89032">
<BindingStrength>5.67</BindingStrength><Concentration
unit="nm">43</Concentration></interaction><interaction
substrate="Sub89033"><BindingStrength>4.37</BindingStrength>
<Concentration unit="nm">75</Concentration></interaction></P><P
id="Mtr245"><interaction substrate="Sub89032">
<BindingStrength>0.65</BindingStrength><Concentration
unit="um">0.53</Concentration></interaction><interaction
substrate="Sub80933"><BindingStrength>8.87</BindingStrength>
<Concentration
unit="nm">8.4</Concentration></interaction></P><S
id="Sub89032"/><S id="Sub89033"/></Array>
```

```
<Array>
   <Protein id="Mas375">
      <interaction substrate="Sub89032">
         <BindingStrength>5.67</BindingStrength>
         <Concentration unit="nm">43</Concentration>
      </interaction>
      <interaction substrate="Sub89033">
         <BindingStrength>4.37</BindingStrength>
         <Concentration unit="nm">75</Concentration>
      </interaction>
   </Protein>
   <Protein id="Mtr245">
      <interaction substrate="Sub89032">
         <BindingStrength>0.65</BindingStrength>
         <Concentration unit="um">0.53</Concentration>
      </interaction>
      <interaction substrate="Sub80933">
         <BindingStrength>8.87</BindingStrength>
         <Concentration unit="nm">8.4</Concentration>
      </interaction>
   </Protein>
   <Substrate id="Sub89032"/>
   <Substrate id="Sub89033"/>
</Array>
```

Ontologies

- The structure of data is its ontology.
  - Database schema
  - XML Document Type Definition (DTD)
- An ontology defines the concepts and relationships between them in a domain.
- Transformations are fundamental:
  - Queries
  - Organizing data (views)
  - Transformation for new purposes

Research Areas

- Ontologies for bioinformatics
- Ontology development in general
  - Constructing ontologies
  - Validation and testing of ontologies
- New ontology languages to capture more meaning
- Transformation languages

Research Areas

- Inference and deduction
  - Logical inference
  - Probabilistic inference
  - Scientific inference
  - Other forms of inference
- Integrating inference with
  - Data mining
  - Experimental processes