

Gene Notebook Example

Mark Graves

*This presentation is Copyright
2001, 2002 by Mark Graves and
contains material Copyright 2002
by Prentice Hall PTR. All rights
reserved.*

Agenda

- **System Requirements**
- **XSL/HTML Implementation**
- **Details**

Gene Notebook

- **Web-based database to organize information about genes**
- **Should be able to add and update information on genes, gene annotations, and gene relationships**

Data Types

- **Gene**
 - Descriptive Information
 - Sequence
 - Sequence Annotations
 - Literature References
- **Pairwise Relationships**
 - Similarity
 - Interaction
- **Relationship Collections**
- **Experimental Data**

Operations

- **Create a gene**
- **Associate information with a gene**
- **Create pairwise relationships and collections of genes**
- **Associate information with those**
- **Search information in database to find its associated gene(s)**

Requirements

- Web-based
- Persistent
- Can be updated

User Interface

The screenshot shows a web browser window with the following content:

Gene Notebook

Genes

BRCA1 Gene 14680

Properties

- organism: Homo sapiens ([Change value](#), [Delete Property](#))
- chromosome_loc: 17q21 ([Change value](#), [Delete Property](#))

Annotations

- DNA_sequence: atggattta ([Change value](#), [Delete Annotation](#))
- protein_sequence: PRKVECNEMDVNGG ([Change value](#), [Delete Annotation](#))

Commands

[Change name of gene](#)

Add property: type: value:

Add annotation: type: value:

Additional Information

[View Microarray Data](#)

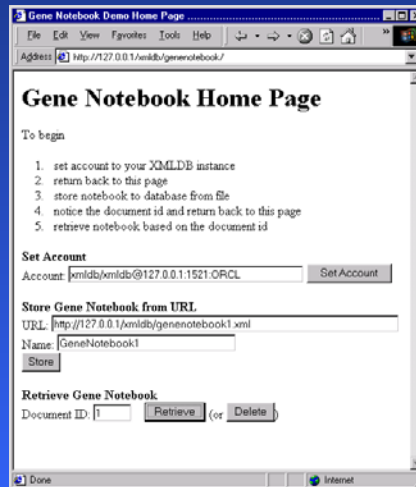
Storage Locations

- **Gene**
 - Descriptive Information (XML DB) ➤
 - Sequence (FASTA) ➤
 - Sequence Annotations (XML DB)
 - Literature References (XML files) ➤
- **Pairwise Relationships (XML Link)**
- **Relationship Collections (XML DB)**
- **Experimental Data (RDB) ➤**

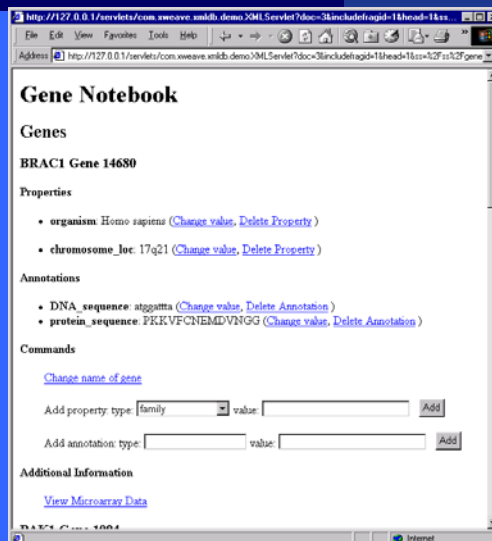
Gene Notebook System

- **HTML Home Page to organize multiple Gene Notebooks**
- **XML Data Server (URL Java Servlet)**
- **XSL Stylesheet (11 templates; 167 lines)**
 - Default XML Templates (node, CData, etc)
 - Main Template (GeneNotebook)
 - Domain-specific Templates (genes, gene, interactions, relationships)

Home Page



Gene Notebook



Gene Notebook



XML

```
<?xml version="1.0"?>
<GeneNotebook>
<genes>
<gene id="14680">
<name>BRCA1</name>
<organism>Homo sapiens</organism>
<annotations>
<DNA_sequence>atggattta</DNA_sequence>
<chr_loc chr="17">17q21</chr_loc>
<protein_sequence>
PKKVFCNEMDVNGG</protein_sequence>
<db_xref gi="555931"/>
</annotations>
</gene>
<gene id="1094">
...
</GeneNotebook>
```



XSL

```
<?xml version="1.0"?>
<!-- Stylesheet for viewing GeneNotebook document -->
<xsl:stylesheet xmlns:xsl="...">
<xsl:template match="/">
<xsl:apply-templates />
</xsl:template>
<xsl:template match="gene">
<h3><xsl:value-of select="name"/> Gene <xsl:value-of
select="@id"/></h3>
<h4>Properties</h4>
<xsl:apply-templates select="*">
<!-- apply templates for sub-elements of gene -->
<xsl:template match="*">
<ul>
<li><b><xsl:node-name/></b>: <xsl:value-of
select="."/>
...
</xsl:stylesheet>
```



Aspects of Gene XSL Template

- Access Gene Properties
- Browse Microarray Data in a RDBMS by Building HREFs in XSL
- Add Property to a Gene
 - Edit XML Fragments
 - Build HTML Forms

Accessing Gene Properties

Properties

- organism: Homo sapiens ([Change value](#), [Delete Property](#))

```
<h4>Properties</h4>
<xsl:apply-templates select="*">
  <!-- apply templates for sub-elements of gene -->
  <xsl:template match="*">
    <ul>
      <li><b><xsl:node-name/></b>: <xsl:value-of select="."/>
        (<xsl:element name="a"><xsl:attribute
          name="href">com.xweave.xmldb.demo.XMLServlet?cmd=store&amp;frag=<xsl:value-of
          select="@FRAGID"/>&amp;texttag=<xsl:node-name/></xsl:attribute>Change
          value</xsl:element>,
          <xsl:element name="a"><xsl:attribute
          name="href">com.xweave.xmldb.demo.XMLServlet?cmd=delete&amp;frag=<xsl:value-of
          select="@FRAGID"/>&amp;</xsl:attribute>Delete Property</xsl:element>
        )</li>
    </ul>
  </xsl:template>
  <xsl:template match="name|annotations">
    <!-- already grapped name, annotations so ignore it this time -->
  </xsl:template>
</xsl:apply-templates>
```

Browsing Relational Data

The screenshot shows a web browser window with two tabs. The active tab displays the 'PAK1 Gene 1094' page, which includes sections for Properties, Annotations, and Commands. The 'Properties' section shows 'organism: Yeast'. The 'Annotations' section shows 'DNA_sequence: ggatttta'. The 'Commands' section includes a 'Change name of gene' link and a form for adding properties or annotations. A second browser window is open in the background, displaying an 'EXPER_RESULT_V LISTING' table.

ID	GENE_ID	NAME	EXPERIMENT	VARIANT	VALUE
3299	1094	PAK1	102	240	0.21
3300	1094	PAK1	102	270	0.18
3301	1094	PAK1	102	300	-0.15
3302	1094	PAK1	102	330	0.19
3303	1094	PAK1	102	360	0.04
3304	1094	PAK1	102	390	0.07
3305	1094	PAK1	103	10	-0.23
3306	1094	PAK1	103	30	-0.25
3307	1094	PAK1	103	50	-0.32
3308	1094	PAK1	103	70	-0.45
3309	1094	PAK1	103	90	-0.67
3310	1094	PAK1	103	110	-0.58
3311	1094	PAK1	103	130	-0.74
3312	1094	PAK1	103	150	-0.64

Building HREFs in XSL

```
<h4>Additional Information</h4>
<div style="margin-left:2em">
<!-- Link to Microarray Data -->
<xsl:element name="a">
  <xsl:attribute name="href">
    com.xweave.xmldb.demo.XMLServlet?
    cmd=rdb&amp;
    name=<xsl:value-of select="name"/>&amp;
    tablename=madb.exper_result_v&amp;
    stylesheet=/ss/generic1.xsl
  </xsl:attribute>
  View Microarray Data
</xsl:element>
</div>
```

Additional Information

[View Microarray Data](#)

```
<a href="com.xweave.xmldb.demo.XMLServlet?
cmd=rdb&name=PAK1&tablename=madb.exper_result_v&stylesheet=/ss/generic1.xsl">
```

Adding Gene Property

ANGPT1 Gene 531

Properties

- organism: Human ([Change value](#), [Delete Property](#))

Annotations

- protein_sequence: MPEPKKVFNCNMDVNGG ([Change value](#), [Delete Annotation](#))

Commands

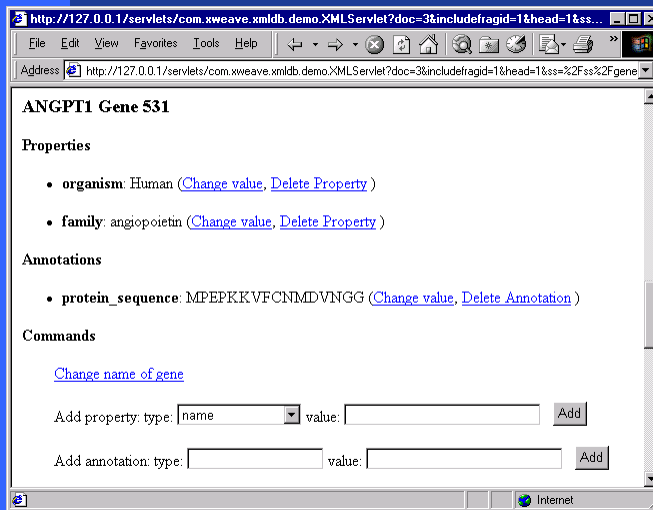
[Change name of gene](#)

Add property: type: value:

Add annotation: type: value:

```
http://127.0.0.1/servlets/com.xweave.xmldb.demo.XMLServlet?
texttag=name&text=angiopoietin&frag=3.15&cmd=Append
```

Protein Family Property Added



XML Data with Fragment Ids

```
<gene FRAGID="3.15" id="531">
  <name FRAGID="3.16">
    ANGPT1
  </name>
  <organism FRAGID="3.17">
    Human
  </organism>
  <annotations FRAGID="3.18">
    <protein_sequence FRAGID="3.19">
      MPEPKKVFCNMDVNGG
    </protein_sequence>
  </annotations>
</gene>
```

```
<gene FRAGID="3.15" id="531">
  <name FRAGID="3.16">
    ANGPT1
  </name>
  <organism FRAGID="3.17">
    Human
  </organism>
  <annotations FRAGID="3.18">
    <protein_sequence FRAGID="3.19">
      MPEPKKVFCNMDVNGG
    </protein_sequence>
  </annotations>
  <family FRAGID="3.30">
    angiotensinogen converting enzyme 1
  </family>
</gene>
```

Building HTML FORMs in XSL

Add property: type: family value: Add

```
<form
  action="com.xweave.xmldb.demo.XMLS
  ervlet" method="GET">
  Add property:
  type:
  <select name="texttag">
    <xsl:for-each
      select="/GeneNotebook/genes/gene/*">
      <option><xsl:node-name/></option>
    </xsl:for-each>
  </select>
  value:
  <input type="input" name="text"
    size="30"/>
    <xsl:element name="input">
      <xsl:attribute name="type">
        hidden</xsl:attribute>
      <xsl:attribute name="name">
        frag</xsl:attribute>
      <xsl:attribute name="value"><xsl:value-of
        select="@FRAGID"/></xsl:attribute>
    </xsl:element>
    <input type="hidden" name="cmd"
      value="Append"/>
    <input type="submit" name="ignore"
      value="Add"/>
  </form>
```

http://127.0.0.1/servlets/com.xweave.xmldb.demo.XMLServlet?
texttag=name&text=angiopoietin&frag=3.15&cmd=Append&ignore=Add