



Bioinformatic Sequence Markup Language

BSML^Ô Reference Manual

June 15, 2002

**Copyright 2001-2002
LabBook, Inc.
All Rights Reserved**

Bioinformatic Sequence Markup Language

Reference Manual

Table of Contents

0. Manual Purpose and Organization.....	1
1. Overview of BSML™	2
1.1 Origins and Goals	2
1.2 Information Encoding	2
1.3 Research Encoding.....	3
1.4 Visualization Encoding	3
1.5 BSML Documents	3
1.6 BSML, Namespaces, and Controlled Vocabularies.....	3
1.7 BSML and MIME Types	3
1.8 BSML and the Genomic XML Viewer.....	4
2. BSML Content	4
2.1 Major Document Sections	4
2.2 Elements Common to Many Content Models.....	5
2.3 Data Representation.....	14
2.4 Research Representation [3].....	39
2.5 Data Visualization.....	43
3. BSML DTD Reference	72
3.1 Parameter Entities	72
3.2 Basic Attributes and Elements	75
3.3 BSML Element	90
3.4 Definitions	91
3.5 Research [3].....	126
3.6 Display.....	138
Index of Elements	181

0. Manual Purpose and Organization

The purpose of this manual is to describe the 2.2 and 3.x (currently 3.1) releases of Bioinformatic Sequence Markup Language (BSML™, copyright LabBook, Inc., 1997-2002). The manual presents the BSML 3.x DTD (Document Type Definition) and includes a good deal of explanatory material to guide the reader's understanding of how BSML may be of use in life science informatics. The manual deals extensively with the representation of biological sequences and the way in which BSML representations relate to other sequence representations (e.g., GenBank flatfile format). Differences between the 2.2 and 3.x versions of BSML are noted.

The original motivation for BSML included creating a sequence representation that would enable graphical rendering of biologically meaningful objects (e.g., a "view" of a sequence and its annotation). It was felt to be equally important to enable the connection of visual display objects to underlying data (e.g., linking the display of a gene to the annotation of that gene). Although BSML may be used strictly for data encoding, the implementation of these additional goals – visualization and linking – requires rendering software. See *Rendering BSML in the Genomic XML Viewer™* for a description of how LabBook's free Genomic XML Viewer™ renders BSML content. See *BSML 2.2 Java API and Toolkit* or the *BSML 3.x Java API and Toolkit* for a discussion of methods and tools for creating BSML content.

This manual is organized in the following sections:

1. **Overview of BSML and the Genomic XML Viewer**
2. **BSML Content** (discussion with examples for each section of the DTD)
3. **Technical Reference** (complete presentation of the DTD)

Typography and references to versions of BSML. In most cases, XML elements are described formally using standard XML DTD notation displayed in a bold fixed-width font. In the body of the text, we refer to elements in a bold font, without tags, e.g., **Sequence** rather than <**Sequence**>. This manual covers both BSML 2.2 and BSML 3.x. BSML 2.2 is a subset of BSML 3.x, and the parts of the DTD that are specific to BSML 3.x are indicated in the DTD presentation as [3].

1. Overview of BSML™

1.1 Origins and Goals

BSML is an XML application created by Joseph Spitzner, Ph. D., Chief Technical Officer of LabBook, Inc. The 2.2 version of BSML is available as a Document Type Definition (DTD) from the LabBook, Inc. website at:

<http://www.labbook.com>

Additional information on BSML may be obtained from

<http://www.bsml.org>

The current draft of **BSML 3.1** is the work of a number of LabBook contributors:

Eluem Blyden
Dean Dai
David Gordon
Chaobo Guo
Seth Kraut
Eric Rentschler
Steven Roggenkamp
Robert Rumpf
Jeffrey Spitzner
Joseph Spitzner

In addition, helpful comment and criticism have been provided by a number of persons from other organizations. The DTD of version 3.1 of BSML is available in draft form by permission of LabBook, Inc. and may be obtained by email request to

bsml@labbook.com

All versions of BSML have been, and will continue to be, released to the public domain. From its initial funding in 1997 by the National Human Genome Research Institute, BSML has been intended as a public standard for the representation of information related to biological sequences.

1.2 Information Encoding

BSML represents a method for encoding information about certain biological molecules (DNA, RNA, and protein sequences), including sequence data, feature tables, references, and associated tabular data (e.g., gene expression values). BSML can represent information from various sources (GenBank, EMBL, Swiss-Prot, etc.). BSML 3.x

provides improved XML content models for describing informational resources and attributing information to persons and organizations.

1.3 Research Encoding

BSML 3.x provides content models for describing various types of research — queries, searches, analyses, and experiments. Research descriptions may be associated with sequence annotation to provide a complete description of the source of evidence for a claim.

1.4 Visualization Encoding

BSML provides a set of **display widgets** that allow display information (e.g., sequence views) to be encoded directly in BSML content. Note that BSML documents do not need to contain display information and may be completely valid and well-formed XML using only the data encoding and/or research portions of BSML.

1.5 BSML Documents

BSML documents may be created in a variety of ways. Most documents are created in response to information management requests, as exports from other applications (e.g., from an SQL-based query utility).

In document form, BSML content may be used to transport information over networks (i.e., as an exchange medium) or to provide persistent content storage. Content may be stored in complete BSML documents (i.e., those containing all sections of a document) or in document fragments that may contain only partial information. Documents of the second type are useful in merging multiple annotations of sequences.

1.6 BSML, Namespaces, and Controlled Vocabularies

BSML attributes are sometimes defined with all values enumerated (e.g., the allowable sequence types). This type of controlled vocabulary is inherent in the DTD itself. In some cases this is either not desirable or feasible, because the controlled vocabulary is very large or unstable. In these instances, mechanisms (links, attributes, or **Authority** [3] elements) are provided for associating a particular term with a particular source, as defined by a **namespace URL**.

1.7 BSML and MIME Types

When BSML content is transported using network protocols such as HTTP, the following MIME types and file extensions are recommended to identify content:

Document Type	File Extensions	MIME Type
Complete BSML document	.bsm, .bsml	text/x-bsml
BSML fragment (overlay)	.bso	text/x-bsov
Zipped BSML documents and supporting files	.bsmz	application/x-bsmz

1.8 BSML and the Genomic XML Viewer

The content of a BSML document may be rendered by suitable software. Because of its use of various linking technologies, BSML content may be visualized in highly interactive ways that allow data mining and drill-down through multiple levels of content. LabBook, Inc. supplies appropriate visualization software in the form of a Genomic XML Browser™ and a Genomic XML Viewer™ (“light” version of the Browser), both of which read complete documents (**.bsm**), document fragments (**.bso**), and zipped documents (**.bsmz**). See *Rendering BSML in the Genomic XML Viewer™* for a complete description of Genomic XML Viewer rendering of BSML.

2. BSML Content

This section reviews BSML representation of various content. The DTD is presented in full in the Reference section (**Section 3**). In this section, only major elements are discussed, and attributes are shown only as required for examples.

XML Style. XML is case-sensitive. BSML *elements* have names that begin with an uppercase letter and are shown in the text below in boldface, e.g., **Bsml**. BSML *attributes* have names that begin with a lower case letter and are shown below in boldface, e.g., **title**. Compound terms for both elements and attributes are generally represented with a hyphen between words, e.g., **Sequence-import**.

2.1 Major Document Sections

A complete BSML document begins with the standard XML processing instruction and encloses all of its content between **Bsml** tags. This element is defined as:

```
<!ELEMENT Bsml
  (Attribute*, Info*, Resource* [3],
   Definitions?,
   Research?[3],
   Display?)
>
```

Example 1. A typical document has the following general structure:

```
<?xml version="1.0"?>
<!DOCTYPE Bsml PUBLIC "-//Labbook, Inc. BSML DTD//EN"
    "http://www.labbook.com/dtd/bsml2_2.dtd">
<Bsml>
...
</Bsml>
```

General information such as document author or creation date may be attached to the entire document by using **Attribute**, **Info**, and **Resource** elements.

Within the **Bsml** tags, the document is divided into three major sections, each of which is optional:

- **Definitions** – genomes[3], sequences, data tables, sets, and networks[3]
- **Research** – search, query, analysis, and experiment descriptions[3]
- **Display** – representations of display widgets for visualization of data

2.2 Elements Common to Many Content Models

Several elements are used in general ways and occur in the content models of many elements.

2.2.1 Attribute, Attribute-List, and Info Elements

Unformatted information that does not belong to one of the explicitly defined elements (e.g., **Sequence**) may be attached to major elements by using **Attribute** elements, which simply define pairs of **name** and **content** attributes.

```
<!ELEMENT Attribute EMPTY >
```

Example 2:

```
<Attribute name="date-created" content="01-MAY-1992 (Rel.22)"/>
<Attribute name="database-xref" content="EMBL:X59720"/>
<Attribute name="SUBCELLULAR LOCATION" content="CYTOPLASMIC"/>
```

If the **content** attribute of an **Attribute** is not defined, the element simply asserts that the **name** attribute is defined. The optional **datatype** attribute may be used to provide datatyping information for the content of an **Attribute** element.

Example 3:

```
<Attribute name="position" content="1001" datatype="integer"/>
```

The **Attribute-list** and **Info** elements are composed of sets of **Attribute** elements. Note: Although these two elements have the same content models, both are present for reasons of backward compatibility with previous implementation of BSML.

```
<!ELEMENT (Attribute-list | Info) (Attribute)* >
```

Example 4:

```
<Attribute-list title="cross-references">
  <Attribute name="database-xref" content="EMBL:X59729"/>
  <Attribute name="database-xref" content="PMID:12345678"/>
</Attribute-list>
```

2.2.2 Resource Identification, Description, and Attribution

In describing annotation, it is useful to be able to associate supplementary informational content, such as bibliographic material or patent citations, with sequence annotation. It is also useful to be able to identify an agent to which content may be attributed, including persons, organizations, software systems, and bibliographic sources. BSML 3.0 introduces the **Resource** element, which is included in the content models of many data and research elements. The **Resource** element includes in its content model the fifteen elements defined for metadata by the **Dublin Core Metadata Initiative**:

<http://dublincore.org>

2.2.2.1 Resource [3]

The **Resource** element has a content model that allows this element to be used quite flexibly to describe a wide variety of resource types. Note that all content models in this section, because they are components of **Resource**, are new to BSML 3.0.

```
<!ELEMENT Resource (Attribute | Coverage | Description | Type |
Relation | Source | Subject | Title | Contributor | Creator | Publisher
| Rights | Date | Format | Identifier | Language | Content | Version |
History | Authority | %links;)* >
```

By itself, a **Resource** element may designate a remote source of information, using the **url** attribute, much like a **Link** element.

Example 5:

```
<Resource title="Analysis Center" url="http://www.analysiscenter.com">
```

2.2.2.2 Creators and Contributors [3]

Typically, an agent is a person, organization, or software system that is connected with a resource through authorship or ownership. At least one of these elements is required by the content model of the **Contributor** and **Creator** elements.


```
<!ELEMENT Creator (Person | Organization | Software-system)+ >
<!ELEMENT Contributor (Person | Organization | Software-system)+ >
```

Example 6:

A resource consisting of a locally stored document might be authored by Jane Doe.

```
<Resource title="Product Description" url="c:\proddesc.doc">
  <Creator>
    <Person lastname="Doe" firstname="Jane"></Person>
  </Creator>
</Resource>
```

2.2.2.3 Person [3]

The **Person** element provides name-based identification of a person, with optional contact information (see **Example 6**).

```
<!ELEMENT Person (Contact-info?) >
```

2.2.2.4 Organization [3]

The **Organization** element identifies an organization by name. Contact information may be supplied using the optional **Contact-info** element, and any number of **Person** elements may be used to identify people associated with the organization.

```
<!ELEMENT Organization (Contact-info?, Person*) >
```

Example 7:

```
<Contributor>
  <Organization name="NCBI"
    description="National Center for Biotechnology Information"
    url="http://www.ncbi.nlm.nih.gov">
    <Person lastname="Darwin" firstname="Charles">
    </Person>
  </Organization>
</Contributor>
```

2.2.2.5 Software System [3]

Software systems may be involved with bioinformatic content in various ways, such as data creation, storage, and retrieval. The **Software-system** element provides the means for describing such agents. Note that the **Version** element (see below) is included in the content model to identify software system versions.

```
<!ELEMENT Software-system
  (Contact-info?, Person*, Organization*, Version*) >
```

Example 8:

```
<Contributor>
  <Software-system name="Oracle"
    description="relational database management system"
    role="data storage">
    <Version major-version="8i" minor-version="3"></Version>
  </Software-system>
</Contributor>
```

2.2.2.6 Contact Information and Postal Address [3]

Contact information may be added to persons, organizations, and software systems and is represented by the **Contact-info** element and the optional **Postal-address** element that it may contain.

```
<!ELEMENT Contact-info (Postal-address?) >
```

```
<!ELEMENT Postal-address EMPTY >
```

Example 9:

```
<Creator>
  <Person lastname="Darwin" firstname="Charles">
    <Contact-info telephone-number="011-555-555-5555">
      <Postal-address
        building-address1="1234 Beagle St."
        country="UK"/>
    </Contact-info>
  </Person>
</Creator>
```

2.2.2.7 Unspecified Content [3]

Occasionally content needs to be included that simply does not fit any of the defined content models or that contains information that should not be parsed. The **Content** element is provided for such cases. The **name**, **description**, **content-type**, and **role** attributes may be used to specify the nature of the content.

```
<!ELEMENT Content (#PCDATA)>
```

Example 10:

```
<Resource title="my password">
  <Content name="mycode.bin">
    <![CDATA[
      AGX09WSDXSDW0834FG0EST
    ]]>
  </Content>
</Resource>
```

2.2.2.8 Version and Date [3]

Version and date information are critical for precisely identifying many resources.

```
<!ELEMENT Version (Organization?, Date?) >
```

```
<!ELEMENT Date EMPTY >
```

Example 11:

```
<Resource>
  <Content>LabBook Enterprise System</Content>
  <Version major-version="2" minor-version="3">
    <Date role="issued" datetime="2001-08-01"/>
  </Version>
</Resource>
```

2.2.2.9 Bibliographic Description [3]

Bibliographic content includes publishers, the description of citations, and the listing of bibliographies. A number of elements support bibliographies. A set of **Resource** elements may define a bibliography.

```
<!ELEMENT Publisher (Person|Organization|Software-system)+ >
```

```
<!ELEMENT Title    (#PCDATA) >
```

```
<!ELEMENT Type    (#PCDATA) >
```

```
<!ELEMENT Identifier (#PCDATA) >
```

Example 12:

```
<Resource>
  <Title>Advances in Biotechnology</Title>
  <Creator>
    <Person lastname="Doe" firstname="John"></Person>
  </Creator>
  <Type>book</Type>
  <Subject>cloning,biotechnology</Subject>
  <Publisher>
    <Organization name="LabBook Press">
      </Organization>
    </Publisher>
  <Date datetime="2001-02-08"/>
  <Identifier context="ISBN" refid="123456789"></Identifier>
  <Content role="abstract">
    Excellent source on new technology
  </Content>
</Resource>
```

2.2.2.10 Patent Citations [3]

Patent-related information may be encoded using the resources and assigning **Attribute** elements to values not represented by other resource elements.

Example 13:

```
<Resource>
  <Type>Patent-citation</Type>
  <Title>Chaetomium xylanase A xlnA gene</Title>
  <Identifier context="USPTO" refid="WO9722692"></Identifier>
  <Date role="issued" datetime="1997-06-26"/>
  <Attribute name="sequence" content="E. coli"/>
</Resource>
```

2.2.2.11 Resource Descriptors [3]

In addition to the elements described above, other elements in the content model of **Resource** are provided so that the content may be characterized in specific ways (see <http://dublincore.org>). Each element in this group has the same general format (see also **Description** in the **Research** section).

```
<!ELEMENT Subject (#PCDATA) >

<!ELEMENT Source (#PCDATA) >

<!ELEMENT Language (#PCDATA) >

<!ELEMENT Coverage (#PCDATA) >

<!ELEMENT Rights (#PCDATA) >

<!ELEMENT Relation (#PCDATA) >

<!ELEMENT Format (#PCDATA) >
```

Example 14a:

```
<Resource>
  <Subject>E. coli mutations</Subject>
  <Source>http://labbook.com</Source>
  <Language>English</Language>
  <Coverage>basic cell function changes</Coverage>
  <Rights>Copyright 2001, LabBook, Inc., All rights reserved</Rights>
</Resource>
```

2.2.2.12 Histories

A **History** element is used to contain a series of **History-event** elements, each of which represents an event in the history that may be characterized by a date, agent, and description. Content internal or external to the current document is associated with the history event by any number of linking elements.

```
<!ELEMENT History (History-event*) >

<!ELEMENT History-event (Date?, Contributor*,
  History-event-description?, %links;) >
```

```
<!ELEMENT History-event-description (#PCDATA) >
```

Example 14b:

```
<Resource>
  <History title="cloning history">
    <History-event title="vector">
      <Date datetime="2001-12-20"/>
      <Contributor>
        <Person fullname="Charles Darwin"></Person>
      </Contributor>
      <History-event-description>
        Preparation of Lambda vector
      </History-event-description>
      <Link title="Lambda" href="SEQ1"/>
    </History-event>
  </History>
</Resource>
```

2.2.2.13 Authority

An **Authority** element is used to identify a particular namespace or ontology (controlled vocabulary) source that applies to an attribute that can take on any set of values (i.e., BSML itself does not specify a controlled vocabulary). The **Authority** element may contain any number of **Authority-domain** elements, each of which specifies the name of an element and one of its attributes.

```
<!ELEMENT Authority (Authority-url*,Authority-domain*) >
```

```
<!ELEMENT Authority-url EMPTY >
```

```
<!ELEMENT Authority-domain EMPTY >
```

Example 14c:

```
<Sequence ... >
  <Feature-tables>
    <Feature-table>
      <Resource>
        <Authority title="NCBI">
          <Authority-url type="website" url="www.ncbi.nlm.nih.gov"/>
          <Authority-domain
            element-name="Qualifier" attribute-name="value-type"/>
        </Authority>
      </Resource>
      <Feature title="Transposon" class="TRANSPOSON">
        <Qualifier value-type="transposon"/>
        <Interval-loc startpos="101" endpos="200"/>
      </Feature>
    </Feature-table>
  </Feature-tables>
</Sequence>
```

2.2.3 Linking

BSML provides several linking mechanisms that associate elements of a document with other BSML elements in the document or with other addressable resources; namely, any resource that may be addressed by a URL.

2.2.3.1 Hierarchical Links

A hierarchical linking mechanism uses implicit links that rest on the hierarchical structure of BSML itself. For example, sequence features are contained in feature tables that are attached to specific sequences. Sequences are thus hierarchically linked to their features.

```
<Sequence>
  <Feature-tables>
    <Feature-table>
      <Feature>
```

2.2.3.2 XML ID Attribute Links

BSML enables each element in a document to define an attribute that creates a unique identifier (**id**="name") for the element (**XML** attribute of type **ID**). Other elements may refer to this identifier in related attributes (**XML IDREF** attributes refer to one identifier, and **IDREFS** attributes refer to a list of identifiers). For example, each **Sequence** element has an **id** attribute, and each **View** element has an attribute named **seqref** that points to the id of the reference **Sequence** for the **View**.

Example 15:

```
<Sequence id="SEQ1" title="SV40">
...
<View seqref="SEQ1" title="SV40 Late">
```

2.2.3.3 Other Attribute Links

Other attributes may also provide linking information. For example, the **dbsource** attribute of a **Sequence** element may point to an external database. Another type of attribute link may be created for elements that can contain **Attribute** elements (generally, all elements in the **Definitions** section of the document). Each **Attribute** element consists of two attributes: **name** and **content**. Such attribute/value pairs may be used to identify other links (e.g., **name**="id" **content**="SEQ1"). Certain **Attribute** elements are expected in association with the **Bsml** element (topmost element) of each document. Some of these may be used for linking to other documents; for example, **<Attribute name**="project", **content**="myproject"/>.

2.2.3.4 Import Links

Several elements allow the import of information from other documents, which may be BSML documents or other types of documents. These elements include **Sequence-import**, **Seq-data-import**, and **Table-import**.

Example 16:

```
<Sequence title="sv40" length="5243"
  representation="raw" molecule="dna">
  <Seq-data-import source="http://www.labbook.com/sv40.8na">
  </Seq-data-import>
</Sequence>
```

2.2.3.5 Explicit Links

Explicit links are defined by using one of the linking elements. **Cross-reference** [3], **Link**, **Extended-link**, and **Group-link**. Note that **Extended-link** elements contain sets of **Locator** elements and **Group-link** elements contain sets of **Document-link** elements. Explicit links contain references to other documents or elements. Each link contains an **href** attribute that specifies a **URL**. The **URL** may refer to any document or BSML element.

BSML linking mechanisms provide methods for navigating from one element to another and from one document to another. A link element may be associated with data elements, display widgets, display pages, or with the complete document.

2.2.3.5.1 Cross-reference Element [3]

```
<!ELEMENT Cross-reference EMPTY > [3]
```

The initial implementation of cross-references in BSML used a **Qualifier** of a **Feature**, with the **value-type** attribute set to **db_xref** and the **value** attribute set to the database and cross-reference identifier. Although this method is still supported in BSML 3.x, its use is **deprecated**. The preferred method is to use the **Cross-reference** element, which allows greater specification of the context and role of the cross-reference.

Example 17:

```
<Cross-reference title="mouse homolog to abc1"
  href="http://www.labbook.com/abc1_mouse.bsm"
  role="mouse homolog"/>
```

2.2.3.5.2 Link Element

The **Link** element is defined as:

```
<!ELEMENT Link EMPTY >
```

Example 18:

```
<Link title="my data" href="http://www.labbook.com/mydata.txt"/>
```

2.2.3.5.3 Extended-link Element

An **Extended-link** element represents a one-to-many linking structure; the element contains any number of **Locator** or **Cross-reference** [3] elements in its content model. The **Extended-link** element is defined as:

```
<!ELEMENT Extended-link (Locator | Cross-reference [3])* >
```

2.2.3.5.4 Locator Element

The **Locator** element has the same attributes as a **Link** element and is defined as:

```
<!ELEMENT Locator EMPTY>
```

2.2.3.5.5 Group-link Element

The **Group-link** element contains any number of **Document-link** elements in its content model.

```
<!ELEMENT Group-link (Document-link)* >
```

2.2.3.5.6 Document-link Element

The **Document-link** element is defined as:

```
<!ELEMENT Document-link EMPTY >
```

2.3 Data Representation

Data are contained in the **Definitions** section of a BSML document, which is defined as:

```
<!ELEMENT Definitions
  Attribute*,
  Genomes?, [3]
  Sequences?,
  Isoforms?, [3]
  Sets?,
  Tables?,
  Networks? [3]) >
```

The general style of this section is to use an element with a plural name (e.g., **Genomes**) to enclose a collection of related elements of one or more types; for example., **Sequences** encloses both the **Sequence** and **Sequence-import** elements.

2.3.1 Genomes [3]

The **Genomes** element defines a collection of **Genome** elements, each of which is defined by a **Genome** element. Note that all genome-specific elements contained in the content model for **Genomes** are new to BSML 3.0.

```
<!ELEMENT Genomes (Attribute*, Genome*, Resource*, %links;) >[3]
```

A genome is typically defined with respect to a particular species (**Organism**, **Organism-clone**, or **Cell-line** element) and usually involves the description of a set of chromosomes. Extrachromosomal genomic DNA may also be defined through the **Extrachromosomal-sequence** element (e.g., mitochondrial, chloroplast, or plasmid DNA). If cytoband location and staining information is available for a chromosome, this content may be added.

```
<!ELEMENT Genome
  (Attribute*, (Organism | Organism-clone | Cell-line)?, Chromosome*,
  Extrachromosomal-sequence*, Resource*, %links;) > [3]
```

```
<!ELEMENT Organism (Attribute*, Description?,
  Strain*, Resource*, %links;) > [3]
```

```
<!ELEMENT Strain (Attribute*, Description?,
  (Organism-clone | Cell-line)*,
  Resource*, %links;) > [3]
```

```
<!ELEMENT Organism-clone (Attribute*, Description?,
  Resource*, %links;) >[3]
```

```
<!ELEMENT Cell-line (Attribute*, Description?, Resource*, %links;) >[3]
```

```
<!ELEMENT Chromosome (Cytoband*, Seq-data-import?)> [3]
```

```
<!ELEMENT Extrachromosomal-sequence (Cytoband*, Seq-data-import?)> [3]
```

```
<!ELEMENT Cytoband EMPTY>[3]
```

Example 19:

```
<Genomes>
  <Genome title="human"
    autosomal-chromosome-count="22"
    sex-chromosome-count="2"
    ploidy-count="2"
    distinct-chromosome-count="24"
    total-chromosome-count="46">
    <Organism name="human" genus="homo" species="sapiens"
      taxon-num="9660">
    </Organism>
    <Resource title="Genome build">
      <Version title="UCSC Golden Path"
        description="bug fix for Oct 2000 release"
        date="2001-1">
      </Version>
    </Resource>
```

```

    <Chromosome name="1" number="1" length="253000000"
        circular="0" autosomal="1">
        <Cytoband band-name="q33.21" major-band="q33" minor-band="21"
            band-color="black" band-type="euchromatin"
            band-start-pos="123456789" band-end-pos="124000000"/>
        ...
    </Chromosome>
    ...
    <Chromosome name="Y" number="24" length="45000000"
        circular="0" autosomal="0">
    </Chromosome>
</Genome>
</Genomes>

```

2.3.2 Sequences

The **Sequences** element, which contains references to any number of DNA, RNA, or protein sequences, is defined as:

```

<!ELEMENT Sequences
    (Attribute*,(Sequence|Sequence-import)*, Segment-set*[3],
    Resource*[3], %links;)>

```

There are two elements that define sequences: **Sequence** and **Sequence-import**. A **Sequence** element defines a sequence in the current BSML document by setting various attributes (e.g., **molecule="dna"** or **topology="circular"**). A **Sequence-import** element uses a URL to refer to another BSML document containing the sequence information. The **Segment-set [3]** element defines a relationship (alignment, translation, etc.) between two or more sequences or sequence fragments.

2.3.2.1 Sequence Element

The **Sequence** element is defined as:

```

<!ELEMENT Sequence (Attribute*,
    Feature-tables?,
    (Seq-data|Seq-data-import)?,
    Numbering?,
    Modification*, [3]
    Resource*, [3]
    %links;)>

```

This content model allows specification of the annotation of the sequence (**Feature-tables**, **Modification**, and **Resource** elements), the sequence data (**Seq-data** or **Seq-data-import** elements), and the position of this sequence in the larger context of a clone or chromosome (**Numbering** element). The attribute list of a **Sequence** is lengthy and allows specification of many characteristics (length, molecule type, strands, etc.) as well as identifiers of the sequence (public and private accession keys).

Example 20:

```
<Sequence      id="P25644"      title="PAT1_YEAST"      molecule="aa"      ic-
acckey="P25644"      length="797"      representation="raw"      topology="linear"
strand="ss" comment="Topoisomerase II-Associated Protein Pat1">
```

2.3.2.2 Sequence Feature Tables

Each **Sequence** may have any number of tables of annotation involving feature tables and feature groups, contained by an element defined as follows:

```
<!ELEMENT Feature-tables (Attribute|Feature-table|Feature-groups[3])* >
```

2.3.2.3 Feature Table

It is common to name feature tables so that each represents a particular group of features (e.g., from a separate analysis). Each feature table may contain any number of features, references, or digest-sets, defined as follows:

```
<!ELEMENT Feature-table
  ((Attribute*,(Reference|Feature|Digest-set)*,Resource*[3],%links;)>
```

2.3.2.4 References

A **Reference** element is defined as:

```
<!ELEMENT                                     Reference
(Attribute*,RefAuthors?,RefTitle?,RefJournal?,Resource*[3],%Links;)>
<!ELEMENT RefAuthors (#PCDATA) >
<!ELEMENT RefTitle   (#PCDATA) >
<!ELEMENT RefJournal (#PCDATA) >
```

Thus each reference may have an author, title, and journal citation. The author, title and journal entities are defined simply as text:

Example 21:

```
<Reference
  dbxref="89017142"
  title="SEQUENCE FROM N.A., AND PARTIAL SEQUENCE"
  comment="SPECIES=Bovine; TISSUE=CEREBELLUM">
<Attribute name="cross-reference" content="MEDLINE=89017142"/>
<RefAuthors>
  Ichimura T., Isobe T., Okuyama T., Takahashi N., Araki K., Kuwano R.,
  Takahashi Y.
</RefAuthors>
<RefTitle>
  Molecular cloning of cDNA coding for brain-specific 14-3-3 protein, a
  protein kinase-dependent activator of tyrosine and tryptophan
  hydroxylases.
</RefTitle>
<RefJournal>
  Proc. Natl. Acad. Sci. U.S.A. 85:7084-7088(1988)
</RefJournal>
```

</Reference>

2.3.2.5 Features

A **Feature** element is defined as shown below:

```
<!ELEMENT                                     Feature
((Attribute*,(%location;|Qualifier)*,Resource*[3],%links;))>
```

where the location entity is defined as

```
<!ENTITY % location "(Site-loc|Interval-loc)">
```

Each feature is identified by its own **title** and lists of zero or more qualifiers and locations. If the feature is a particular **type** (e.g., as specified in a source GenBank document) its value is generally encoded in the **value-type** core attribute.

Note: If a **View** element is associated with the **Sequence** for a **Feature**, rendering software may be instructed to create a display widget (**Interval-widget** or **Point-widget**) automatically for the feature by setting **display-auto="1"**.

Example 22:

```
<Feature id="FTR4" title="gene" class="GENE" value-type="gene"
  display-auto="1">
  <Qualifier value-type="gene"/>
  <Interval-loc startpos="1513" endpos="2373" complement="1"/>
</Feature>
```

2.3.2.6 Feature Qualifiers

A **Qualifier** element is defined as shown below:

```
<!ELEMENT Qualifier (ATTRIBUTE*,RESOURCE*,%LINKS;) >
```

Qualifiers generally have a type and value, which are encoded in the **value-type** and **value** attributes.

Example 23:

```
<Qualifier value-type="organism" value="Cloning vector pAP3neo"/>
<Qualifier value-type="db_xref" value="taxon:69403"/>
```

2.3.2.7 Feature Locations

Two types of location element are defined. A single position on a sequence is shown by a **Site-loc** element, defined as:

```
<!ELEMENT Site-loc (ATTRIBUTE*,RESOURCE*,%LINKS;) >
```

If the site is on the complementary strand, **complement="1"** is added (another attribute is available to indicate that the strand is unknown). If **complement="0"** or the **complement** attribute is omitted, the site is assumed to lie on the coding strand (for sites on sequences for which this designation is appropriate, e.g., a non-palindromic pattern match).

Example 24:

```
<Site-loc sitepos="100001" complement="1"/>
```

The second type of location represents a region on the sequence:

```
<!ELEMENT Interval-loc (Attribute*,Resource*,%links;) >
```

If **startopen="1"** then the location of the starting position is assumed to lie at or upstream of **startpos**. Similarly, if **endopen="1"** then the location of the end position is assumed to lie at or downstream of **endpos**. If **onepos="1"** then the location is assumed to be a single position that lies within the range specified by **startpos** and **endpos**. If the range is on the complementary strand, **complement="1"** is added. If **complement="0"** or the **complement** attribute is omitted, the location is assumed to lie on the coding strand, for ranges on sequences for which this designation is appropriate (e.g., a gene).

Example 25a:

```
<Interval-loc startpos="1001" endpos="1200" complement="1"/>
```

If a **Feature** contains a single location specifier, that specifier defines the position of the **Feature** completely. Some features, however, consist of a number of separated regions (e.g., a **JOIN** of several exons to define the coding regions of a gene). To capture this information, the **join** attribute [3] is set to true (**join="1"**). In this case, the first **Interval-loc** element identifies the region spanned by all of the joined segments and each particular segment is identified by the succeeding **Interval-loc** elements.

Example 25b:

```
<Feature id="FT4" title="gene" class="GENE" value-type="gene" join="1">
  <Qualifier value-type="gene"/>
  <Interval-loc startpos="1513" endpos="2373"/>
  <Interval-loc startpos="1513" endpos="1713"/>
  ...
  <Interval-loc startpos="2201" endpos="2373"/>
</Feature>
```

Note: In BSML 2.2 it is assumed that a join is represented in the same way if there is more than one location element, even though there is no **join** attribute.

2.3.2.8 Digest Sets

A **Digest-set** is a set of one or more **Restriction-enzyme** elements, each of which contains sites for a particular restriction enzyme. The **Digest-set** element is defined as follows:

```
<!ELEMENT Digest-set (Attribute*,(Restriction-enzyme)*,  
                      Resource*[3],%links;) >
```

The title of the digest set, if any, is specified in the **title** attribute. When the element is contained by a **Sequence** element, the attributes for sequence descriptors need not be specified (**seqref**, **seqname**, **is-circular**). If the search for restriction sites was limited to a portion of the sequence, the values for **start-pos** and **end-pos** may be supplied; otherwise it is assumed that the complete sequence was searched. If results are included only for restriction enzymes with a number of cuts within an allowable range, the values for **min-cuts** and **max-cuts** may be provided.

Usually, the restriction site data are defined by entering sites for each enzyme; the sizes of the fragments are then computed. However, there are cases in which the data consist of fragment sizes. In this case, the **fragments** attribute, which consists of a comma-separated list of fragment sizes, may be supplied as illustrated in the following example:

Example 26:

```
fragments="1000,5000,10000"
```

The **units** attribute is only provided if the **fragments** attribute is defined, and it is used to specify the units (**kb**, **mb**, etc.) that apply to the fragment sizes.

Restriction-enzyme elements are defined as follows:

```
<!ELEMENT Restriction-enzyme EMPTY >
```

The name of the restriction enzyme is specified using the **title** attribute (e.g., **title="EcoRI"**). When the data are included as part of a **Sequence**, the **source** attribute is usually omitted. The **recognition-seq** attribute is used to specify the recognition sequence for the restriction enzyme (e.g., **recognition-seq="accggt"**). The cleavage position from the start of the recognition sequence is indicated by **offset="n"**, where **n** is the number of bases 3' of the start of the sequence (e.g., if the pattern is **acc^ggt**, **offset="3"**). The **overhang** is set to the 5' overhang produced by the enzyme cut, with **0** indicating a blunt cutter. The **sites** are entered as a comma-separated list of positions (e.g., **sites="1,501,4334,2009"**). It is assumed that each site number refers to the position immediately 3' of the location on the sequence at which the enzyme cleaves the sequence.

Example 27:

```
<Digest-set id="DGS1" title="SphI">  
  <Restriction-enzyme title="SphI" source="ECRPOBC"  
    recognition-seq="GCATGC" offset="5" overhang="-4"  
    sites="628,6926,8710"/>  
</Digest-set>
```

2.3.2.9 Feature Group [3]

A **Feature-group** element defines a set of related features that are contained in **Feature** elements in a **Feature-table**. For example, this element may be used to relate a set of exons for a gene (in the case where the exons are entered as separate features rather than being encoded as a join). Each group contains any number of **Feature-group-member** elements, each of which points to a **Feature**.

```
<!ELEMENT Feature-group (Attribute*, Feature-group-member*,
    Resource*, %links;)> [3]
```

```
<!ELEMENT Feature-group-member (#PCDATA) > [3]
```

Example 28:

```
<Feature-group group-set="SET1">
  <Feature-group-member featref="FTR1" feature-type="EXON"
    group-type="Splice-variant">
</Feature-group-member>
  <Feature-group-member featref="FTR2" feature-type="EXON"
    group-type="Splice-variant">
</Feature-group-member>
  Exons included in this splice variant
</Feature-group>
```

2.3.2.10 Sequence Data

There are two ways by which sequence data (bases or residues) may be associated with a **Sequence**. A **Seq-data** element is used to enter the data in the current BSML document as part of the sequence definition. A **Seq-data-import** element is used to access the data from an external file or URL.

2.3.2.11 Sequence Data in the Document

A **Seq-data** element has no attributes and is defined as follows:

```
<!ELEMENT Seq-data (#PCDATA) >
```

Data must be in text format using IUPAC single character codes (IUPACna for DNA or RNA sequence and IUPACaa for protein sequences). The sequence data are simply included as follows.

Example 29:

```
<Seq-data>
  acgtacgt
</Seq-data>
```

When the sequence is read, characters that are not in the IUPAC character set are ignored, including numbers and control characters (e.g., line feed characters). Thus the following is acceptable.

Example 30:

```
<Seq-data>
  1 acgtacgtac gtacgtactt acgtcgcgaa cgccgtaact aaaacccctt
  51 acgtacgtac acgtacgtac acgtacgtac acgtacgtac ggatatcgtc
</Seq-data>
```

2.3.2.12 Imported Sequence Data

A **Seq-data-import** element is defined as follows:

```
<!ELEMENT Seq-data-import (Attribute*) >
```

In addition to an ASCII text file, the format may be one of the following (as defined by NCBI):

```
IUPACna: IUPAC 1 letter codes nucleotide codes, no spaces
IUPACaa: IUPAC 1 letter amino acid codes, no spaces
NCBI2na: 4 bases per byte with 2 bits per residue, 00=A,01=C,10=G,11=T
NCBI4na: 2 bases per byte with 4 bits for each for residue,
          0001=A, 0010=C, 0100=G, 1000=T/U
          0101=Purine, 1010=Pyrimidine, etc.
```

Example 31:

```
<Seq-data-import format="IUPACaa"
  source="c:\labbook\genomicbrowser\library\lib1\query.fcgi">
</Seq-data-import>
```

In BSML 3.0, **start-pos** and **length** attributes are defined for **Seq-data-import** that permit access to sequence data from a larger sequence store.

Example 32:

```
<Seq-data-import source="u01317.8na" format="IUPACna"
  start-pos="101" length="200" >
</Seq-data-import>
```

2.3.2.13 Importing a Sequence

The **Sequence-import** element is defined as follows:

```
<!ELEMENT Sequence-import (Attribute*,Resource*[3],%links;)>
```


The **source** attribute is a reference to the BSML file containing the **Sequence** to be imported. The **identifier**, if present, selects one sequence from a set (if there is only one sequence in the document, the **identifier** is not needed).

Example 33:

```
<Sequence-import source="www.labbook.com/myseq.bsml" identifier="SEQ1">
```

2.3.2.14 Relative Sequence Numbering Basis

Internally, the length of every sequence is expressed as a positive integer. Sequences and other sequence-related elements may specify their unit of measurement (bp, cM, etc.) through the use of a **Numbering** element, which also may define a relative numbering basis or a linear transformation rule for converting internal sequence positions to display values. By using a reference attribute (**seqref**), a **Numbering** element may indicate the location of a sequence in a larger context, e.g., the location of a sequence on a clone or the location of a clone on a chromosome. By chained references, a sequence may be placed in more than one positional context.

```
<!ELEMENT Numbering EMPTY >
```

The following example illustrates relative numbering that assigns 39037 to the first base of the sequence.

Example 34:

```
<Numbering      use-numbering="1"      type="continuous"      has-zero="0"
ascending="1" refnum="39037"/>
```

The following example illustrates how a sequence may be located on the complementary strand of a clone, which, in turn, is located on the coding strand of a chromosome.

Example 35:

```
<Sequence title="Chromosome 1" id="CHR1"
      segmenttype="chromosome"
      representation="virtual" length="253000000">
</Sequence>
<Sequence title="Clone 1" id="CLONE1"
      segmenttype="clone"
      representation="virtual" length="200000">
  <Numbering seqref="CHR1" type="continuous"
    refnum="143000001" ascending="1"/>
</Sequence>
<Sequence title="Sequence 1" id="SEQ1"
      segmenttype="sequence"
      representation="raw" length="50">
  <Numbering seqref="CLONE1" type="continuous"
    refnum="123456" ascending="0"/>
<Seq-data>
```

```

    acgt...
  </Seq-data>
</Sequence>

```

2.3.2.15 Modifications [3]

Each **Sequence** may include a list of modifications, where each modification indicates a change to a base or residue. The **change** attribute may be used as a controlled vocabulary relative to a namespace or standard defined in the optional **source** attribute.

```
<!ELEMENT Modification (Attribute*,Resource*,%links;) > [3]
```

Example 36a:

```
<Modification position="1001" change="methylation" source="ST.25"/>
```

2.3.2.16 Segment-sets [3]

A **Segment-set** refers to any number of **Segment** elements, each of which refers to a **Sequence**. Collectively the set establishes a relationship between the sequences that may reflect an assembly or an alignment.

```
<!ELEMENT Segment-set (Attribute*,Segment+,Resource*,%links;) > [3]
```

```
<!ELEMENT Segment (Attribute*,Resource*,%links;) > [3]
```

Example 36b:

```

<Segment-set seg-set-type="alignment">
  <Segment seg-id="SEQ1" seg-start="1001" seg-end="2000"/>
  <Segment seg-id="CLONE1" seg-start="25403" seg-end="26402"/>
</Segment-set>

```

2.3.3 Isoforms [3]

Isoform elements are used to define particular changes in sequences related to alleles, SNPs, and mutations. To allow for various ways of specifying the position of a change, the attribute **location**, defined as character data, is used rather than an integer positional reference. This allows positions to be specified in general ways and in any relevant manner (e.g., in a cytoband). Another element in this section, **Case**, is used to refer to a set of changes associated with a particular organism (patient, clone, etc.). The **Phenotype** element is used to describe phenotypes associated with particular isoforms or cases. The **Pedigree-set** element contains references to any number of **Pedigree** elements. These may group **cases** by lineage and indicate standard pedigree information.

Note: All elements in this section are new as of BSML 3.x.

```

<!ELEMENT Isoforms (Isoform-set | Case-set | Phenotype-set |
                    Genotype-set | Pedigree-set)* > [3]

```

```

<!ELEMENT Isoform-set (Attribute*,
                      (Organism | Organism-clone|Cell-line)?,
                      Isoform*, Resource*, %links;) > [3]

<!ELEMENT Isoform (Attribute*, Resource*, %links;) > [3]

<!ELEMENT Phenotype-set (Attribute*,
                       (Organism | Organism-clone | Cell-line)?,
                       Phenotype*, Resource*, %links;) > [3]

<!ELEMENT Phenotype (Attribute*, Resource*, %links;) > [3]

<!ELEMENT Case-set (Attribute*,
                   (Organism | Organism-clone | Cell-line)?,
                   Case*, Resource*, %links;) > [3]

<!ELEMENT Case (Attribute*, (Organism | Organism-clone | Cell-line)?,
               Isoform*, Life-event*, Genotype-set*,
               Resource*, %links;) > [3]

<!ELEMENT Life-event EMPTY > [3]

<!ELEMENT Genotype-set (Attribute*,
                      (Organism | Organism-clone | Cell-line)?,
                      genotype*, Resource*, %links;) > [3]

<!ELEMENT Genotype (Haplotype*) > [3]

<!ELEMENT Haplotype EMPTY > [3]

<!ELEMENT Pedigree-set (Attribute*, Pedigree*, Resource*, %links;)> [3]

<!ELEMENT Pedigree (Attribute*,
                  (Organism | Organism-clone |Cell-line)?,
                  Mating*, Resource*, %links;) > [3]

<!ELEMENT Mating (Life-event?,Children?) > [3]

<!ELEMENT Children (Case*) > [3]

```

Example 37:

```

<Isoforms>
  <Isoform-set>
    <Isoform id="ISO1001" seqref="SEQ1" phenoref="PHEN1"
            title="mutation 168B"
            location-type="physical" location="1001"
            change="cac" replaces "tgt">
    </Isoform>
  </Isoform-set>
  <Phenotype-set>
    <Phenotype id="PHEN1">
      Extra ear
    </Phenotype>
  </Phenotype-set>
</Isoforms>

```

Example 38:

```
<Isoforms>
  <Case-set>
    <Case title="John Doe" isoforms="ISO1001"></Case>
  </Case-set>
</Isoforms>
```

2.3.4 Sets

The **Sets** element is defined as follows:

```
<!ELEMENT Sets (Attribute*, Set*, Resource*[3], %links;) >
```

Thus the **Sets** element is simply a list of any number of **Set** elements, each of which is defined as follows:

```
<!ELEMENT Set (Attribute*, List-member*[3], Resource*[3], %links;) >
```

The **Set** element itself consists of a list of space-separated **id** references to BSML elements in the current document. Each **id** may point to any valid element (e.g., a **Sequence**) that has an **id** attribute.

Example 39:

```
<Sets>
  <Set idrefs="SEQ1 SEQ2 SEQ3"></Set>
</Sets>
```

A **Set** may also include any number of **List-member**[3] elements:

```
<!ELEMENT List-member EMPTY > [3]
```

The **source** attribute of a **List-member** may be used to link the member to other objects, including elements in the current documents.

Example 40:

```
<Sets>
  <Set>
    <List-member id="COL1001" title="DNA Sequences"/>
    <List-member id="COL1002" title="Protein sequences"/>
  </Set>
</Sets>
```

2.3.5 Data Tables

The **Tables** element is defined as follows:

```
<!ELEMENT Tables (Attribute |
                  Table |
                  Motif-table |
                  Sequence-search-table [3] |
                  Multiple-alignment-table [3] |
                  Seq-pair-alignment |
                  Alignment-point-set |
                  PCR-primer-table [3] |
                  Table-import)* >
```

A variety of elements may be included in this section, each of which describes a particular type of data table, hierarchy, or set of data objects. Some of the elements in this section are derived from public domain sources.

2.3.5.1 HTML Table

The BSML definition for a **Table** follows the html 4.0 specification. Adding HTML tables to BSML documents is complicated and seldom necessary. In most cases, if the table already exists, it is simpler to save the HTML table to a file and bring its contents into BSML by a **Table-import** element.

2.3.5.2 Importing a Table

The **Table-import** element is defined as follows:

```
<!ELEMENT Table-import (Attribute*, Table-data*,
                        Resource*[3], %links;) >
```

The following example imports data from a text table with fixed-width columns. One column contains text (start at position 11 on each line and extend for 3 characters). The other column contains numeric data (start at position 12 and extend for 7 characters). The data are read from a local file named **ltaghydr.dat**.

Example 41:

```
<Table-import id="TIM1" title="hydrophobicity" format="TXT"
  column-headers="aa,value" column-starts="11,22" column-widths="3,7"
  header-count="0" display-widths="0.5in,1in"
  alignments="L,R" font="FNT2"
  add-row-numbers="1" locked-columns="0" source="ltaghydr.dat">
</Table-import>
```

Normally the data for the table are obtained from the external source defined in the **source** attribute (hence the name **Table-import**). The data may, however, be contained in the document as a set of rows of comma-separated values. Each row is defined in a **Table-data** element:

```
<!ELEMENT Table-data EMPTY >
```

2.3.5.3 Sequence-search-table [3]

This element, along with an appropriate **Analysis** element defined in the **Research** section, relates to results of a sequence search. The results may be contained in an external file, in which case the **url** attributes points to that file (e.g., a file containing results reported in the NCBIBLAST XML format). Alternatively, the results may be passed internally in the **Sequence-search-output** element, using a CDATA section as necessary.

```
<!ELEMENT Sequence-search-table (Attribute*,
                                (BlastOutput | Search-output), Resource*, %links;)> [3]
```

```
<!ELEMENT Sequence-search-output (#PCDATA) > [3]
```

Example 42:

```
<Sequence-search-table search-type="NCBIBLAST" url="blast.xml"
  queryseqref="SEQ1" analysisref="ANA1">
</Sequence-search-table>
```

2.3.5.4 Multiple Alignment [3]

This table type represents multiple alignments using formats most closely associated with CLUSTALW, a freely available multiple alignment program:

Thompson J.D., Higgins D.G., Gibson T.J.; "CLUSTAL W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice."; Nucleic Acids Res. 22:4673-4680(1994).

For more information, see:

ClustalW WWW Service at the European Bioinformatics Institute
<http://www2.ebi.ac.uk/clustalw>
Rodrigo Lopez, Services Programme

```
<!ELEMENT Multiple-alignment-table (Attribute*, CLUSTAL-parameters?,
  Alignment-summary?, Pairwise-alignments?,
  Alignment-groups?, Sequence-alignment*, Resource*, %links;) > [3]
```

```
<!ELEMENT CLUSTAL-parameters EMPTY > [3]
```

```
<!ELEMENT Alignment-summary (Aligned-sequence*) > [3]
```

```
<!ELEMENT Aligned-sequence EMPTY > [3]
```

```
<!ELEMENT Pairwise-alignments (Aligned-pair*) > [3]
```

```
<!ELEMENT Aligned-pair EMPTY > [3]
```

```
<!ELEMENT Aligned-groups (Aligned-group*, Total-alignment) > [3]
```

```
<!ELEMENT Aligned-group EMPTY > [3]
```

```

<!ELEMENT Total-alignment EMPTY > [3]

<!ELEMENT Sequence-alignment (Sequence-data*,Alignment-consensus) > [3]

<!ELEMENT Sequence-data (#PCDATA) > [3]

<!ELEMENT Alignment-consensus (#PCDATA) > [3]

```

Example 43:

```

<Multiple-alignment-table>
  <CLUSTAL-parameters
    title="-NONE-" type="Protein" method="full" output=""
    outorder="aligned" ktuple="0" window="5" score="percent"
    topdiags="0" pairgap="0" matrix="blosum" gapopen="0"
    endgaps="0" gapext="0" gapdist="0">
  </CLUSTAL-parameters>
  <Alignment-summary seq_type="Protein" seq_format="Pearson">
    <Aligned-sequence seq="1" name="FOSB_HUMAN" length="395">
    </Aligned-sequence>
    ...
  </Alignment-summary>
  <Pairwise-alignments>
    <Aligned-pair pair="1:2" score="85"></Aligned-pair>
    ...
  </Pairwise-alignments>
  <Alignment-groups groups="3">
    <Aligned-group group="1" total_seq="2" group_score="5773"/>
    <Total-alignment align_score="11049"/>
  </Alignment-groups>
  <Sequence-alignment sequences="4">
    <Sequence-data seq_name="FOSB_MOUSE">
      MFQAFPGDYDSGSRCSS-----
    </Sequence-data>
    ...
    <Alignment-consensus>
      *****
    </Alignment-consensus>
  </Sequence-alignment>
</Multiple-alignment-table>

```

2.3.5.5 PCR Primers [3]

The elements defined here are closely associated with the parameters, inputs, and outputs used with the public domain PRIMER3 program (Steve Rozen, Helen J. Skaletsky; copyright 1996,1997,1998 Whitehead Institute for Biomedical Research. All rights reserved).

For more information see

http://www-genome.wi.mit.edu/genome_software/other/primer3.html

```

<!ELEMENT PCR-primer-table (Attribute*, PCR-primer*, PCR-summary?,
    PCR-statistical-analysis?, Resource*, %links;) > [3]

<!ELEMENT PCR-primer (Primer-sequence*, PCR-product?) > [3]

<!ELEMENT Primer-sequence EMPTY > [3]

<!ELEMENT PCR-summary EMPTY > [3]

<!ELEMENT PCR-product EMPTY > [3]

<!ELEMENT PCR-statistical-analysis (PCR-statistics*) > [3]

<!ELEMENT (PCR-statistics) EMPTY > [3]

```

Example 44:

```

<PCR-primer-table>
  <PCR-primer>
    <Primer-sequence primer-left="1" start="179" length="20"
      tm="60.09" gc_percent="50.00" any="2.00" three_end="0.00"
      seq="CCACCATCACCATTACCACA"/>
    <Primer-sequence primer-left="0" start="383" length="20"
      tm="60.21" gc_percent="45.00" any="5.00" three_end="2.00"
      seq="CTGGCATTGCTTTCCAGAAT"/>
  </PCR-primer>
  <PCR-summary seq_size="519" include_region_size="519"/>
  <PCR-product product_size="205" pair_any_compl="4.00"
    pair_3_compl="1.00"/>
</PCR-primer>
...
<PCR-statistical-analysis>
  <PCR-statistics statistics-left="1" considered="3413"
    too_many_Ns="0" in_target="0" in_excl_reg="0"
    bad_GC_percent="2" no_GC_clamp="0" tm_too_low="649"
    tm_too_high="1977" high_any_compl="0" high_3_compl="7"
    poly_X="46" high_end_stab="60" ok="672"/>
  <PCR-statistics statistics-left="0" considered="3396"
    too_many_Ns="0" in_target="0" in_excl_reg="0"
    bad_GC_percent="2" no_GC_clamp="0" tm_too_low="505"
    tm_too_high="2197" high_any_compl="3" high_3_compl="14"
    poly_X="44" high_end_stab="62" ok="569"/>
</PCR-statistical-summary>
</PCR-primer-table>

```

2.3.5.6 Motif Tables

A motif table contains data related to one or more sequences. A **motif** is a characteristic of a sequence defined by one or more sequence features and the relations among these features. A motif is present for a sequence if the qualifications for the individual features and their relationships are met. For example, a two-element motif might consist of an **Sp1** site upstream of a **start codon** that is located within 100 bases of the **Sp1** site. The results

of simple searches may be represented as one-element motifs (e.g., a list of **prosite** pattern matches defines a one-element motif).

BSML requires that all motif elements be defined, even if they are simply named **Element 1**, **Element 2**, etc. That is, elements are referred to by their titles and each element refers to successive members of each motif. This does not present serious problems, because "empty" elements may be included (essentially, fillers). Thus, if a motif consists of elements **A**, **B**, and **C**, one instance of the motif might be **[100, 200, 300]**, indicating that element **A** is at position 100, **B**, at 200, and **C**, at 300. Another instance might be **[252, 0, 500]**, indicating that motif element **B** is missing (a **position** of **0** indicates a missing member). In addition, all motifs do not need to have the same number of elements (e.g., one motif may consist of **[100, 500, 800]** and the next may consist of **[150, 535]**, indicating that there is no member for the third element; in effect, including less than the total number of members is equivalent to adding members with **position=0**.

The following terminology is used below:

- a. **Element**: The description of one component of the complete motif (e.g., the first element is an **Sp1** site)
- b. **Motif**: A particular instance consisting of positions for each element **[100, 200, 300]**
- c. **Member**: The position for a particular element in a particular motif (e.g., the second member of the motif **[100, 200, 300]** is **200**)
- d. **Empty members**: Undefined elements that are indicated by a position of **0**

2.3.5.6.1 Element Types

Some elements refer to a specific position on the sequence and some do not.

2.3.5.6.1.1 Positional Elements

Positional elements are associated with a location or region on the sequence. If an element is positional, it has a location specified as a base (residue) **position** > 0. To indicate a region (e.g., a match from a BLAST search), the element may also have a **width** (number of bases), which is 1 by default.

2.3.5.6.1.2 Nonpositional Elements

Nonpositional elements are not associated with any particular location or region of a sequence (or the positional information is not of interest). Any value < 0 indicates a nonpositional element. The particular number (-1, -2, etc.) has meaning if state names are defined for the element (see below). For example, a gene expression level might be associated with a motif that indicates (through its positional members) the locations of mutations.

2.3.5.6.2 Additional Characteristics

2.3.5.6.2.1 Motifs

Each motif may have two additional characteristics: **title** and **value** (integer). If these values are not stated explicitly for the overall motif, the title and value of the first member of the motif are taken as the title and value for the overall motif. A motif **title** is used if there is some meaningful identity for the motif (e.g., **title="patient 23"**). A motif **value** is used if there is some overall index associated with the motif as a whole (rather than the individual members); e.g., **value="18"** might indicate the survival time in months for **patient 23**.

2.3.5.6.2.2 Members

Members may have a number of optional characteristics associated with them (in addition to **position** and **width**):

- **Title**: a caption for this particular member (e.g., the **element** might be a restriction enzyme site, and a **particular member** might have the title **EcoRI**)
- **Value**: an integer indicating some quantitative characteristic of the member
- **Strand**: location on the plus or minus strand
- **Marker**: offset from the 5' end of the region of the member (e.g., a cleavage site)
- **Overhang**: 5' overhang on the minus strand, to locate a marker on the minus strand

2.3.5.6.3 Motif Encoding in BSML

BSML encodes a motif using the **Motif-table** element. A particular motif table may contain data for any number of sequences, but the motif elements must be the same for all motifs for all sequences within a particular table. If more than one sequence is involved, an overall sequence length (**global-length**) may be defined, and each sequence is then aligned to a position on the global sequence.

```
<!ELEMENT Motif-table (Attribute*, Numbering?, Interval-loc*,
    Motif-element*, Sequence-motif*, Resource*[3], %links;) >
```

Most of these attributes relate to motif visualization.

Example 45:

```
<Motif-table
    display-gap="0.04in" display-height="0.2in" id="MTVS1234554"
    point-width="0.2in" title="NM_002893 (Sequence): aatc">
  <Attribute content="aatc" name="Pattern Name"/>
  <Attribute content="aatc" name="Pattern"/>
  <Attribute content="None (mismatch cost=1)" name="Cost Matrix"/>
  <Attribute content="No" name="Allow Indels"/>
  <Attribute content="0" name="Maximum Cost"/>
  <Attribute content="0" name="Minimum Positions"/>
```

```

<Attribute content="0" name="Marker"/>
<Attribute content="0" name="5' Overhang"/>
<Attribute content="both" name="Strand"/>
<Attribute content="all" name="Frame"/>
<Motif-element border-color="blue" fill-bg-color="aqua"
    fill-fg-color="red" fill-pattern="fdiag" title="aatc"/>
<Sequence-motif endpos="1946" seqref="G13259504" startpos="1">
  <Motif-data one-element="1"
    positions="350,465,513,644,648,690,786,802,1128,1225"
    strands="1,0,1,1,1,1,1,0,1,1">
    </Motif-data>
  </Sequence-motif>
</Motif-table>

```

2.3.5.6.3.1 Attribute

Attribute elements are used optionally to supply additional name/content pairs to describe the overall motif, for example, search conditions.

2.3.5.6.3.2 Numbering

The optional **Numbering** element provides a relative numbering basis for displaying positions on the global sequence.

2.3.5.6.3.3 Interval-loc

The optional **Interval-loc** elements are used to supply markers on the global sequence for reference purposes.

Attributes

- **display-auto** is set to true (1) to create automatic display information for the sequence motif.
- **auto-view** is set to true (1) to create a **View** element automatically.
- **seqref** is a required attribute that is the **id** of the reference sequence (e.g., **id="SEQ0"**).
- **alignment** is the position on the global sequence at which this sequence is aligned (e.g., **alignment="1000"**).
- **startpos** is the starting position on the sequence for the region containing motif members (e.g., **startpos="5001"**).
- **endpos** is the ending position on the sequence for the region containing motif members (e.g., **endpos="10000"**).
- **refs** is a space-separated list of **id** references.

2.3.5.6.3.4 Motif-element

The **Motif-element** entries define the motif elements. Each element is defined by a **Motif-element** that describes many properties used in controlling the visualization. The

core attribute **title** is used to specify the name of the element (e.g., **title="Sp1"**).

```
<!ELEMENT Motif-element EMPTY >
```

2.3.5.6.3.5 Sequence-element

The data for each sequence are grouped under a **Sequence-element**.

2.3.5.6.3.6 Sequence-motif

The actual motif data are organized in terms of individual sequences. Each **Sequence-motif** element must have a **seqref** attribute that refers to a **Sequence** defined in the **Sequences** section of the document. The region of the sequence containing the motifs is indicated by the **startpos** and **endpos** attributes. If there is more than one sequence in the overall motif table, each sequence may be aligned to a particular base (residue) on the global sequence (the default is position 1). This feature is indicated by the **alignment** attribute.

```
<!ELEMENT Sequence-motif (Interval-loc*, Motif-data*) >
```

For example, the following definition indicates that the data derive from the sequence with **id="SEQ0"** and the range of interest is from bases 101 to 300. This sequence is aligned to position 501 of the global sequence.

Example 46:

```
<Sequence-motif      seqref="SEQ0"      startpos="101"      endpos="300"  
alignment="501">
```

The optional **Interval-loc** elements are used to supply markers on the sequence for reference purposes.

2.3.5.6.3.7 Motif-data

The contents of the members are defined by **Motif-data** elements.

```
<!ELEMENT Motif-data (Motif-data*) >
```

2.3.5.6.5 Motif Encoding

There are several ways in which data may be encoded for a motif.

2.3.5.6.5.1 Single-element Encoding

Each instance of **Motif-data** encodes more than one instance of a motif consisting of a single element. To indicate this encoding, set **one-element="1"**. For example, the following encoding defines four instances of a one-element motif.

Example 47:

```
<Motif-data one-element="1" positions="100,200,300,400"></Motif-data>
```

If all attributes were defined for this example, the record might appear as follows.

Example 48:

```
<Motif-data one-element="1"
  positions="100,200,300,400"
  widths="6,8,4,6"
  strands="1,0,0,1"
  titles="EcoRI,HindIII,BamHI,EcoRI"
  values="25,72,58,37"
  markers="2,4,2,2"
  overhangs="-2,2,0,-2"
  refs="FTR1"></Motif-data>
```

2.2.5.6.5.2 Flat Encoding

In **flat** mode, each instance of **Motif-data** encodes one instance of a multi-element motif (**one-element** is undefined or set to "0"). For example, the following encoding defines one instance of a three-element motif.

Example 49:

```
<Motif-data positions="100,200,300"></Motif-data>
```

2.3.5.6.5.3 Hierarchical Encoding

In **hierarchical** mode, instances of **Motif-data** are parents of nested child instances. The highest level encodes the first member of the motif, etc. For example, the following encoding represents four instances of a three-element motif (indentation is not required and is used only to make the nesting clear).

Example 50:

```
<Motif-data position="100">
  <Motif-data position="200">
    <Motif-data position="300"></Motif-data>
    <Motif-data position="400"></Motif-data>
  </Motif-data>
  <Motif-data position="600">
    <Motif-data position="800"></Motif-data>
    <Motif-data position="900"></Motif-data>
  </Motif-data>
</Motif-data>
```

This is equivalent to the flat representation:

```

<Motif-data positions="100,200,300"></Motif-data>
<Motif-data positions="100,200,400"></Motif-data>
<Motif-data positions="100,600,800"></Motif-data>
<Motif-data positions="100,600,900"></Motif-data>

```

Note also that the final child level (i.e., the level having no children) may have data for more than one motif, as though **one-element="1"** were defined at that level. The following example encodes the same data as above.

Example 51:

```

<Motif-data position="100">
  <Motif-data position="200">
    <Motif-data position="300,400"></Motif-data>
  </Motif-data>
  <Motif-data position="600">
    <Motif-data position="800,900"></Motif-data>
  </Motif-data>
</Motif-data>

```

2.3.5.7 Sequence Pair Alignments

A **Seq-pair-alignment** element represents the relationship between fragments of two sequences.

```

<!ELEMENT Seq-pair-alignment (Attribute*, Seq-pair-run*,
    Resource*[3], %links;) >

```

In addition to the description conveyed by these attributes, individual runs of similarity may be defined by the following element:

```

<!ELEMENT Seq-pair-run (Attribute*, Resource*[3], %links;) >

```

Example 52:

```

<Seq-pair-alignment id="SPA1" refseq="S1"
  refcaption="sv40" refstart="1"
  refend="5243" reflength="5243" compseq="S2" compcaption="polyoma"
  compstart="1" compend="5243" complength="5243" method="dotplot"
  runminscore="0" runmaxscore="100" totalscore="20">
  <Seq-pair-run runlength="86" refpos="575"
    comppos="539" runscore="2">
  </Seq-pair-run>
  <Seq-pair-run runlength="500" refpos="1000"
    comppos="1100" runscore="10">
  </Seq-pair-run>
  <Seq-pair-run runlength="400" refpos="2100"
    comppos="1900" runscore="20">
  </Seq-pair-run>
  <Seq-pair-run runlength="350" refpos="2410"
    comppos="2610" runscore="5">
  </Seq-pair-run>

```

```

    <Seq-pair-run runlength="61" refpos="4109"
      comppos="3958" runscore="0">
    </Seq-pair-run>
    <Seq-pair-run runlength="1287" refpos="3100"
      comppos="3300" runscore="30">
    </Seq-pair-run>
  </Seq-pair-alignment>

```

Note: Additional attributes have been added in BSML 3.0 to allow for improved representation of alignments.

2.3.5.8 Alignment Point Sets

In some cases there is a point of alignment to some topic of interest that relates two or more sequences to each other. To describe such a relationship, use the **Alignment-point-set** element.

```

<!ELEMENT Alignment-point-set (Attribute*, Numbering?, Resource*,
%links;) >

```

Example 53:

```

<Alignment-point-set seqids="V3 V4 V5 V6 V7 V8 V9"
  featids="52 52 52 52 52 52 52"
  numseg="1"
  starts="1,1,1,1,1,1,1"
  seglens="159" >
</Alignment-point-set>

```

Note that this data structure is **deprecated** and duplicates some of the functionality of the new **Segment-set** and **Segment** elements.

2.3.6 Networks [3]

A **Network** element provides a representation of a directed or undirected graph in which the connectors between nodes may be weighted or not. The link elements in the content models of the set of elements allow them to be connected in other ways to other elements and resources.

Note: All of the elements in this group are new as of BSML 3.0.

```

<!ELEMENT Networks (Network*) > [3]

<!ELEMENT Network (Attribute*, Node*, Arc*, Network-selection-set*,
  Resource*, %links;) > [3]

<!ELEMENT Node (Attribute*, Resource*, %links;) > [3]

<!ELEMENT Arc (Attribute*, Resource*, %links;) > [3]

<!ELEMENT Network-selection-set EMPTY > [3]

```

Example 54:

```
<Networks>
  <Network id="NET1" title="The Network" directed="1" weighted="0">
    <Node id="NODE1" title="Node 1" node-type="branch"></Node>
    <Node id="NODE2" title="Node 2" node-type="leaf"></Node>
    <Node id="NODE3" title="Node 3" node-type="leaf"></Node>
    <Arc sourcenode="NODE1" destnode="NODE2"></Arc>
    <Arc sourcenode="NODE1" destnode="NODE3"></Arc>
  </Network>
</Networks>
```


2.4 Research Representation [3]

The elements in this section encode a variety of research content, where research includes both experimental and informatics-based activities. Note that all of the elements in this section are new as of BSML 3.0.

The organizing element for this main BSML section is defined as:

```
<!ELEMENT Research
  (Attribute*, Searches?, Queries?, Analyses?, Experiments?) > [3]
```

Some of the elements in this section record actions by using the **Log-entry** element, which is defined as:

```
<!ELEMENT Log-entry (#PCDATA | Measure | Measure-range)* > [3]
```

2.4.1 Searches

The **Searches** element encloses any number of **Search** elements and is defined as:

```
<!ELEMENT Searches (Attribute*, Search*)> [3]
```

2.4.2 Search

```
<!ELEMENT Search (Attribute*, Search-conditions?, Search-result-map?,
  Search-result-set*, Resource*, %links;) > [3]
```

```
<!ELEMENT Search-conditions (#PCDATA) >[3]
```

```
<!ELEMENT Search-result-map (Search-result-field-desc*)>
```

```
<!ELEMENT Search-result-field-desc EMPTY>
```

```
<!ELEMENT Search-result-set (Search-result-item*)>
```

```
<!ELEMENT Search-result-item (Search-result-field*,
  (Link | Cross-reference)*)>
```

```
<!ELEMENT Search-result-field (#PCDATA)>
```

Example 55:

```
<Searches>
  <Search id="SCH1">
    <Search-conditions>
      database=GenBank
      keyword=topoisomerase
    </Search-conditions>
  </Search>
</Searches>
```

2.4.3 Queries

The **Queries** element encloses any number of **Query** elements and is defined as:

```
<!ELEMENT Queries (Attribute*, Query*) > [3]
```

2.4.4 Query

Each **Query** element specifies the query applied to a data set (typically as a set of SQL statements) and may contain a description of the results returned.

```
<!ELEMENT Query (Attribute*, Query-request*, Query-return*,
                 Resource*, %links;) > [3]
```

```
<!ELEMENT Query-request (#PCDATA) > [3]
```

```
<!ELEMENT Query-return (#PCDATA) > [3]
```

Example 56:

```
<Queries>
  <Query id="QRY1" database="MYDATA">
    <Query-request request-type="SQL">
      <![CDATA[
        SELECT * FROM MYTABLE
      ]]>
    </Query-request>
    <Query-return>
      <![CDATA[
        Processed as XML
      ]]>
    </Query-return>
  </Query>
</Queries>
```

2.4.5 Analyses

The **Analyses** element encloses any number of **Analysis** elements and is defined as:

```
<!ELEMENT Analyses (Attribute*, Analysis*) >[3]
```

2.4.6 Analysis

Each **Analysis** typically specifies input parameters and input data. The description of the analysis is often linked by **id** to a data table or other result set.

```
<!ELEMENT Analysis (Attribute*, Parameter*, Input-data*,
                   Resource*,%links;) > [3]
```

```
<!ELEMENT Parameter EMPTY > [3]
```

```
<!ELEMENT Input-data (#PCDATA) > [3]
```

Example 57:

```
<Analyses>
  <Analysis id="ANA1" title="hydrophobicity analysis">
    <Input-parameter name="window-size" value="25" type="int"/>
    <Link title="sequence" href="#SEGMENT1"/>
  </Analysis>
</Analyses>
```

2.4.7 Experiments

The **Experiments** element encloses any number of **Experiment** elements and is defined as:

```
<!ELEMENT Experiments (Attribute*, Experiment*) > [3]
```

2.4.8 Experiment

```
<!ELEMENT Experiments (Attribute*, Experiment*)> [3]
```

```
<!ELEMENT Experiment (Attribute*, Description?,
  (Enzyme | Equipment | Materials | Note |
   Product | Reagent | Sample | Solution |
   Structure | Supplier | Terminology)*,
  Protocol*, Resource*, %links;)> [3]
```

```
<!ELEMENT Description (#PCDATA | Log-entry |
  Measure | Measure-range | Link)* > [3]
```

```
<!ELEMENT Log-entry (#PCDATA | Measure | Measure-range)* > [3]
```

```
<!ELEMENT Measure-range (Measure+) > [3]
```

```
<!ELEMENT Measure (Measure-quantity, Measure-units?) > [3]
```

```
<!ELEMENT Measure-quantity (#PCDATA) > [3]
```

```
<!ELEMENT Measure-units (#PCDATA) > [3]
```

```
<!ELEMENT Enzyme (Attribute*, Description?, Resource*, %links;)> [3]
```

```
<!ELEMENT Equipment (Attribute*, Description?, Resource*, %links;)> [3]
```

```
<!ELEMENT Note (Attribute*, Description?, Resource*, %links;)> [3]
```

```
<!ELEMENT Materials (Attribute*, Description?, Resource*, %links;)> [3]
```

```
<!ELEMENT Product (Attribute*, Description?, Resource*, %links;)> [3]
```

```
<!ELEMENT Reagent (Attribute*, Description?, Resource*, %links;)> [3]
```

```
<!ELEMENT Sample (Attribute*, Description?,
  (Organism | Organism-clone | Cell-line)?,
  Development-stage?, Tissue-sample*,
```

```

Resource*, %links;)> [3]

<!--ELEMENT Organism (Attribute*, Description?, Strain*,
Resource*, %links;)> [3]

<!--ELEMENT Strain (Attribute*, Description?,
(Organism-clone | Cell-line)*,
Resource*, %links;)> [3]

<!--ELEMENT Organism-clone (Attribute*, Description?,
Resource*, %links;)> [3]

<!--ELEMENT Cell-line (Attribute*, Description?, Resource*, %links;)> [3]

<!--ELEMENT Development-stage (#PCDATA) > [3]

<!--ELEMENT Tissue-sample (#PCDATA) > [3]

<!--ELEMENT Solution (Attribute*, Description?, Resource*, %links;)> [3]

<!--ELEMENT Structure (Attribute*, Description?, Resource*, %links;)> [3]

<!--ELEMENT Supplier (Attribute*, Description?, Resource*, %links;)> [3]

<!--ELEMENT Terminology (Attribute*, Description?,
Resource*, %links;)> [3]

<!--ELEMENT Protocol (Attribute*, Description?, Protocol-step*,
Resource*, %links;) > [3]

<!--ELEMENT Protocol-step (Attribute*, Description*,
Resource*, %links;)> [3]

```

Example 58:

```

<Experiments>
  <Experiment>
    <Reagent value-type="solvent">
    </Reagent>
    <Protocol id="PRT1">
      <Protocol-step>
        <Description>SDS</Description>
      </Protocol-step>
    </Protocol>
  </Experiment>
</Experiments>

```

2.5 Data Visualization

The **Display** section of a BSML document conveys information for rendering the visualization of various display **widgets**. Note that the rendering examples shown below were produced using the Genomic XML Viewer, available free from LabBook, Inc. For more information, see *Rendering BSML in the Genomic XML Viewer*.

2.5.1 Page-oriented Media

BSML display is defined for a page-oriented medium, and the display section may contain any number of **Page** elements. All display widgets are contained on pages.

2.5.2 Bio-widget Characteristics

Each display widget has characteristics that are specific to its function (e.g., draw a chart). Most are specified in terms of the position, size, and color of basic drawing elements (lines and text). Most display widgets are selectable and many have links to underlying data defined in the **Definitions** section.

2.5.2.1 Page Positioning and Size

2.5.2.1.1 Units of Distance for Expressing Length

Locations on the page and distance values such as line lengths are specified in terms of horizontal and vertical coordinates, expressed as a **%length** attribute value. Distance and location may use a variety of measures:

px = pixels

% = percentage of the page width (length)

cm = centimeters

in = inches

The default measurement unit is set in an attribute of the **Display** element (e.g., **default-unit="cm"**). This unit is supplied if no unit is specified (e.g., **hloc="10"**). It is permissible to specify horizontal and vertical coordinates of a point in different dimension units, for example, **hloc="10.2cm" vloc="20%"**.

If a length is followed by **"@idV:"** and an object's **id**, the value is taken from a location or size of the corresponding widget. **"V"** is one of the following:

x : horizontal center

y : vertical center

w : width

h : height

l : left

r : right

t : top
b : bottom

For example, if a sequence view-line with **id="VLN1"** has a horizontal center of 300 pixels, a horizontal length of **"20px@idx:VLN1"** sets the value to 320 pixels. This method is most commonly used in resolving coordinates on the page.

2.5.2.1.2 Units of Line Width

Line width is specified in an attribute with the value **%linewidth**, which may be expressed as a relative or absolute value, or by using one of a set of enumerated types:

- Absolute value units include pixels (**px**), inches (**in**), and centimeters (**cm**); e.g., **line-width="0.01in"**.
- Relative values are stated without units as values between **1** (thinnest) and **7** (thickest). The value of the thinnest line is specified as an attribute of the **Display** element (e.g., **line-width-base="0.01in"**). The increment applied from one step to another is specified as an attribute of the **Display** element (e.g., **line-width-inc="0.005in"**). Thus the absolute value of a relative value is calculated by adding a multiple of the **line-width-inc** to the **line-width-base** (e.g., **3=line-width-base+2*line-width-inc**).
- Enumerated values selected from **"thin","medium","thick"**, which correspond to relative values of **1**, **2**, and **4**, respectively.

2.5.2.2 Color

An attribute value type "color" (**%color**) may be either a hexadecimal number (prefixed by a hash mark) or one of the following 16 color names:

Black	= "#000000"	Green	= "#008000"
Silver	= "#C0C0C0"	Lime	= "#00FF00"
Gray	= "#808080"	Olive	= "#808000"
White	= "#FFFFFF"	Yellow	= "#FFFF00"
Maroon	= "#800000"	Navy	= "#000080"
Red	= "#FF0000"	Blue	= "#0000FF"
Purple	= "#800080"	Teal	= "#008080"
Fuchsia	= "#FF00FF"	Aqua	= "#00FFFF"

Thus the color values **"#800080"** and **"Purple"** both refer to the color purple.

2.5.3 Utility Display Elements

A number of elements in the **Display** section use the same utility elements in their content models. These elements are described here.

2.5.3.1 Quantifier Display Format Control

A **Quantifier** controls a display feature of a line (**width**, **color hue**, or **color saturation**) used in drawing a display widget. The input to the element is a numerical **value** that is used to compute an output level, which controls the value of the line characteristic. The **Quantifier** element is defined as follows:

```
<!ELEMENT Quantifier EMPTY >
```

The **type** attribute specifies which line characteristic is to be controlled. If the **type** is either **saturation** or **length**, the **base** and **slope** values are used to compute the output value. If the **type** is **color**, the **red-**, **blue-**, and **green-****base** and **-slope** are used to compute the separate rgb components needed to define the color. If the **type** is **saturation**, the rgb component to be controlled is selected as the **val-name** (e.g., **val-name="blue"**).

The **minimum** and **maximum** values set the lower and upper limits for output values. Both **min-val** and **max-val** are used in the computation; if **x** is the input value, the value used in computation is calculated as:

$$\text{value} = (\text{x-min-val}) / (\text{max-val} - \text{min-val})$$

The appropriate formula is then applied (i.e., **output = base + value * slope**, in the case of width and saturation values). For example, if a chart plots lines using saturation proportional to length, the following definition might be included.

Example 59:

```
<Quantifier type="SATURATION" min-val="20.0" max-val="30.0"  
  minimum="50" maximum="255" val-name="red" base="0" slope="255">
```

This definition means that the input value **x** (which in this example lies in the range 20-30) is used to calculate the **values** with the formula:

$$\text{value} = (\text{x}-20.0) / (30.0-20.0)$$

The result of this calculation is then entered into the formula:

$$\text{output} = 0 + 255 * \text{value}$$

The result is compared to a maximum of **255** and a **minimum** of **50** to determine the saturation of the line color. For example, if **x = 25**, **value = 0.5**, and **output = 127.5**,

which is the value returned as the saturation for the color component selected as **val-name** (="red" in the example).

2.5.3.2 Page Location Coordinate

Most display widgets are located on the page by a **Coord** element that specifies the horizontal (**hloc**) and vertical (**vloc**) positions on the page of the upper left corner of the widget.

```
<!ELEMENT Coord EMPTY >
```

The coordinate values may be specified in any of the methods available for specifying **length**.

2.5.3.3 Page Margin

A **Margin** element is used to set the boundaries of a page or display region. The margin values may be specified in several ways, depending on which attributes are defined:

```
<!ELEMENT Margin EMPTY >
```

- To specify each value individually, define values for **left**, **right**, **top**, and **bottom**.
- To specify the same value for the left and right margins, set **horizontal**.
- To specify the same value for the top and bottom margins, set **vertical**.
- To specify the same value for all four margins, set **margin**.

2.5.3.4 Monitor Display Control

The **Screen** element may be attached to the **Display** element or to a **Page**. When attached to **Display**, the element sets the default page attributes to be used with all **Page** elements that do not have their own **Screen** elements. When attached to a **Page** element, the **Screen** element sets the page layout attributes to be used with that page only.

```
<!ELEMENT Screen (Margin?) >
```

- **width** sets the width of the drawing canvas, exclusive of printing margins in the case of printed output.
- **height** sets the height of the drawing canvas, exclusive of printing margins in the case of printed output.
- **color** sets the background color for monitor output and defaults to **white**.
- **monochrome** sets whether monitor output should be color or black and white.

If the **Screen** element has an optional **Margin** element defined, the margin sets limits on the area used to display objects (i.e., defines a clipping region).

2.5.3.5 Printed Output Display Control

The **Paper** element may be attached to the **Display** element or to a **Page**. When attached to **Display**, the element sets the default printer output attributes to be used with all **Page**

elements that do not have their own **Paper** elements. When attached to a **Page** element, the **Paper** element sets the printer output attributes to be used with that page only.

```
<!ELEMENT Paper (Margin?) >
```

- **monochrome** defines whether output should be in color or black and white.
- **landscape** defines the paper orientation, which defaults to the current printer setting if this attribute is not defined.

If a **Margin** element is defined, the margin defines a clipping region on the printed page.

2.5.3.6 Widget Border Format Control

A **Border** element is used to print a rectangle enclosing a display widget.

```
<!ELEMENT Border EMPTY >
```

1. **Line-width** defines the width of the border line.
2. **Line-color** defines the color of the plotted line.
3. **Gutter** defines an indent from the border line to the area surrounded by the border.

2.5.3.7 Point Symbol Display Control

A **Symbol** element defines the shape and size of a symbol (circle, square, or triangle) to be used in plotting a point on a sequence line.

```
<!ELEMENT Symbol EMPTY >
```

2.5.3.8 Axes

The **X-axis** and **Y-axis** elements define the display characteristics of an axis used to show positions on a sequence.

```
<!ELEMENT (X-axis|Y-axis) (Numbering?) >
```

2.5.4 Primary Elements of the Display Section

The **Display** section of a BSMML document is defined as follows:

```
<!ELEMENT Display (Links?,Fonts?,Styles?,Screen?,Paper?,Page+) >
```

Relative line width values are stated without units as values between 1 (thinnest) and 7 (thickest). The value of the thinnest line equals the **line-width-base**. The increment applied from one step to another equals **line-width-inc**. Thus the absolute value of a relative value is calculated by adding a multiple of the **line-width-inc** to the **line-width-base** (e.g., **3=line-width-base+2*line-width-inc**). The enumerated values "thin", "medium", and "thick" correspond to relative values of 1, 2, and 4, respectively.

2.5.5	Document	Level	Links
--------------	-----------------	--------------	--------------

The **Links** element of the **Display** section defines document level links that may be accessed by **id** from any element in the document. This element is simply defined as follows and may contain any number of any of the elements defined in the **links** entity:

```
<!ELEMENT Links %links;>
```

2.5.6 Fonts

The attributes of the **Fonts** element define document-wide default fonts to be used in displaying text and numerical values. The content model consists of a list of any number of **Font** elements.

```
<!ELEMENT Fonts (Font*) >
```

2.5.6.1 Default Fonts

Default fonts are used for any font attribute that is not specified for a display widget. For example, if a **Caption-widget** does not define its **font** attribute, the **text-font** is used to display the caption. If any specific default font (**base-**, **text-**, **numb-**, or **fixed-font**) is undefined, default values supplied by the software are used. All of these fonts are defined by reference to the **id** of one of the **Font** elements.

2.5.6.2 Font Sizes

Font widths may be specified in points, e.g., **size="12pt"**. Alternatively, the width may be specified as an absolute number between **1** and **7**. In the latter case, the values assigned by the **font-sizes** attribute determine the font size in points. These values are entered as a comma-separated list, e.g., **font-sizes="6,8,10,12,14,18,24"**. In this example, setting **size="3"** for a font is equivalent to **size="10pt"**.

2.5.6.3 Font Elements

Bsml **Font** elements are adapted from the html 4.0 specification and are defined as:

```
<!ELEMENT Font EMPTY >
```

2.5.7 Styles

BSML uses the **Cascading Style Sheet** (CSS 2.0) specification to provide general formatting for all element attributes of a particular type or a particular **class**. The **Styles** element is used to include style information either directly, using the **Style** element, or by reference to a CSS style document, using one or more **Style-import** elements.

```
<!ELEMENT Styles (Style-import*,Style?) >
```

```
<!ELEMENT Style-import EMPTY >
```

```
<!ELEMENT Style (#PCDATA) >
```

Example 60:

```
<Styles>
  <Style-import source="http://labbook.com/mystyle.css"/>
  <Style type="text/css">
    view-line-widget {plus-color:"red"; line-width:"0.02in";}
    interval-widget {border-color:"blue"; fill-bg-color:"yellow";}
  </Style>
</Styles>
```

2.5.8 Screen Control

The **Screen** element of the **Display** section is optional; if present, it describes display page layout information to be used as the default for any **Page** element that does not have its own **Screen** element. If no **Screen** element is present, the rendering software provides default values.

2.5.9 Pages

The **Paper** element of the **Display** section is optional; if present, it describes printer output information to be used as the default for any **Page** element that does not have its own **Paper** element. If no **Paper** element is present, the rendering software provides default values. Any number of **Page** elements may be included in the **Display** section. The **Page** element is defined as follows:

```
<!ELEMENT Page (Screen?, Paper?, Border?,
  ( %pagedisplaywidgets; | %setwidgets; | View )*,
  Object?, Resource*, %links;) >
```

As the content model indicates, each **Page** may have a **Screen**, **Paper**, and **Border** element associated with it. Also, a **Page** may have any number of **Link** elements associated with it. The remainder of the content model for a **Page** consists of the following:

1. **View** elements that provide views of individual sequences and their features
2. **pagedisplaywidgets**, which consist of "stand-alone" display widgets such as captions, lines, and shapes
3. **setwidgets**, which consist of composite display elements that involve references to more than one sequence or feature

2.5.10 Display Widgets

A display widget is a graphic object that may be rendered on the display page or in its own window. Some display widgets are simple and have no subcomponents. Other

display widgets are complex and may have other display widgets or subcomponents attached to them.

Rendering note: When a display widget is plotted, one or more **selection rectangles** are recorded for that element. If the display widget is simple, one rectangle is recorded that contains the plotted figure. If the display widget is complex, a number of rectangles may be recorded that correspond to the locations of subcomponents of the complete widget. When a simple widget is selected with the mouse (by clicking on it), a click anywhere on the widget has the same effect. When a complex widget is selected, subcomponents may be selected by clicking on a particular part of the widget.

2.5.11 Sequence-independent Display Widgets

Sequence-independent widgets are those that can be defined without reference to a **Sequence** element in the **Definitions** section. Some of these widgets can be displayed only on the page canvas, whereas others may be shown in their own windows. Some may be displayed in either way.

2.5.11.1 Text Display Widgets

The following widgets are used for displaying text.

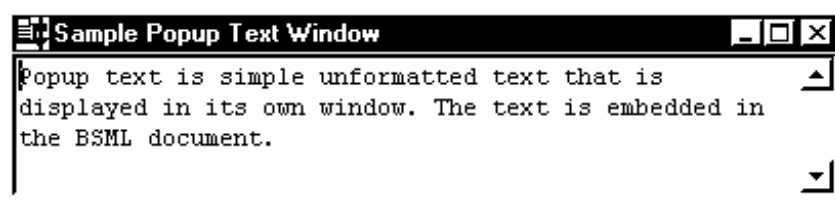
2.5.11.1.1 Popup-text-widget

A **Popup-text-widget** is used to display unformatted text in a popup window of its own. This widget is usually used to provide explanatory text that is displayed by activating a link.

```
<!ELEMENT Popup-text -widget (#PCDATA) >
```

Example 61:

```
<Popup-text-widget id="POP1" title="Sample Popup Text Window">  
  Popup text is simple unformatted text that is displayed in its own  
  window. The text is embedded in the BSML document.  
</Popup-text-widget>
```



The popup window, which appears as shown above, might be activated by clicking on a caption defined as follows:

Example 62:

```
<Caption-widget text="View popup text window" onclick="LINK1">
  <Coord hloc="0.1n" vloc="0.9in" >
    <Link id="LINK1" title="Popup Text Window" href="#POP1">
</Caption-widget>
```

Note that the **title** attribute of the **Popup-text-widget** is used as the caption for the stand-alone window in which the text is displayed. The text is word-wrapped to the current size of the window.

2.5.11.1.2 Caption-widget

A **Caption-widget** is used to display text on a page. The text may be formatted in a variety of ways and is printed using the font referred to by the **font** attribute.

```
<!ELEMENT Caption-widget (Coord, Resource*, %links;) >
```

A variety of examples is shown below to illustrate the effects of different settings.

2.5.11.1.2.1 Caption widget - horizontal text, single line

Example 63:

```
<Caption-widget text="View popup text window">
  <Coord hloc="1.0n" vloc="0.9in">
    <Link id="LINK1" title="Popup Text Window" href="#POP1">
</Caption-widget>
```

2.5.11.1.2.2 Caption widget - horizontal text, word-wrapped

To create word-wrapped text, set **wordwrap="1"** and set **wrap-length** to the maximum width of the text on one line.

Example 64:

```
<Caption-widget text="To view references, click on sv40 and then click
on the book icon on the toolbar (if no book icon appears, click the
circle first)" font="FNT5" wordwrap="1" wrap-length="2in">
  <Coord hloc="0.2in" vloc="3.0in">
</Caption-widget>
```

```
To view references, click on
sv40 and then click on the book
icon on the toolbar (if no book
icon appears, click the circle
first)
```

2.5.11.1.2.3 Caption widget - aligning text

If the text is drawn horizontally (the default state), the text may be aligned to the **left** (default), **center**, or **right** by setting the **align** attribute equal to one of these values. For example, the last example appears as shown below if **align="right"**.

Example 65:

To view references, click on
sv40 and then click on the book
icon on the toolbar (if no book
icon appears, click the circle
first)

Note that the horizontal location now sets the **right** edge of the aligned text. Similarly, if **align="center"**, the **hloc** attribute of the **coord** element sets the center alignment point.

2.5.11.1.2.4 Caption widget - vertical text

Vertical text is drawn with the letters in the upright position. This text is created by setting **orientation** to **down**.

Example 66:

```
<Caption-widget text="vertical" orientation="DOWN">
  <Coord hloc="5.84in" vloc="1.20in">
</Caption-widget>
```

c
a
p
t
i
o
n

2.5.11.1.2.5 Caption widget - angled text

Angled text is created by setting the **orientation** attribute to **angled** and setting **angle** to the desired angle in degrees (0=horizontal). Negative angles cause the text to be inverted.

Example 67:

```
<Caption-widget text="45 angle" orientation="ANGLED" angle="45">
  <Coord hloc="6.10in" vloc="1.20in">
</Caption-widget>
```

45 angle

2.5.11.1.2.6 Caption widget - bordered text

To draw a border line around the text, set **border="1"** and then set the **border-gutter** (distance from text to border line), **border-color** (color of border line), and **border-width** (thickness of border line).

Example 68:

```
<Caption-widget text="Linking and Launching Examples" font="FNT14"
  border="1" border-gutter="0.1in" border-width="0.05in"
  border-color="red">
  <Coord hloc="0.3in" vloc="2.8in">
</Caption-widget>
```



2.5.11.1.2.7 Caption widget - underlined text for links

In keeping with the usual display mode in HTML browsers, it is generally a good idea to display captions that activate links as underlined text (that is, users will expect the text to activate a link based upon their browser experience). This is simply a matter of setting the **font** attribute to refer to a font that has **underline="1"** in its definition.

2.5.11.2 Lines and Shapes

2.5.11.2.1 Line-pointer-widget

A Line-pointer-widget plots a line that:

1. May be solid, dashed, or dotted (**line-style** attribute).
2. May be single-segmented or multi-segmented (number of **Coord** elements attached to the line pointer).
3. May have an arrowhead drawn on either end (or both ends) (**use-start-arrow** and **use-end-arrow** attributes).
4. May have angled text centered along its length (**caption** attribute) that may be displayed above or below the line (**above-line** attribute).

The **Line-pointer-widget** is defined as follows:

```
<!ELEMENT Line-pointer-widget (Coord+, Resource*, %links;) >
```

The **line-width** and **line-color** attributes control the thickness and color of the line. If an arrowhead is drawn at the start, its size is controlled by **start-arrow-length** and **start-arrow-width**; this is also true for an arrowhead drawn at the end of the line.

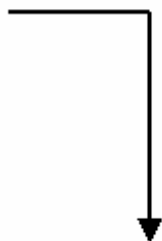
Example 69:

```
<Line-pointer-widget line-style="solid" line-width="0.02in"
```

```

line-color="black" use-start-arrow="1"
start-arrow-length="0.10in" start-arrow-width="0.10in">
<Coord hloc="5.00in" vloc="1.30in"/>
<Coord hloc="5.00in" vloc="0.36in"/>
<Coord hloc="4.44in" vloc="0.36in"/>
</Line-pointer-widget>

```



2.5.11.2.2 Shape-widget

A **Shape-widget** defines the display conditions for a **shape** that may be set to **ellipse**, **rectangle**, or **rounded-rectangle** (these are the permitted values defined by the **displayshape** entity). Regardless of its shape, the shape is enclosed in a rectangle set by the **height** and **width** attributes. If **filled="1"**, the shape is drawn as a solid using the specified **color**. If **filled="0"**, the shape is drawn unfilled using a line of the specified **color** and of thickness **line-width**. The **Shape-widget** is defined as shown below:

```
<!ELEMENT Shape-widget (Coord, Resource*, %links;)>
```

Example 70:

```

<Shape-widget height="0.10in" width="0.10in" filled="1"
  line-width="0.01in" shape="ellipse" color="black">
  <Coord hloc="4.98in" vloc="1.34in"/>
</Shape-widget>

```



2.5.11.2.3 Axis-widget

An **Axis-widget** is a stand-alone axis that displays numerical scaling information on an abscissa (if **X-axis** defined) or ordinate (if **Y-axis** defined). The numbering displayed on the axis is controlled by an (optional) **Numbering** element attached to the **x-** or **y-axis** element. An axis may have the following features:

1. Each axis represents an integer range from the required **start** to the required **end** value.
2. Numbering may proceed from either end of the axis line, depending on whether **ascending** is true (1).
3. A **caption** may be displayed using the **cap-font** font either above (right) or below (left) of the axis, depending on the value of **topright**.

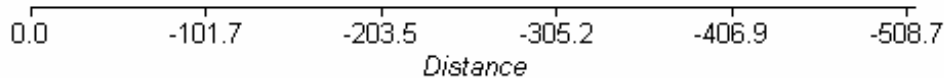
4. The length of the axis line is set by **axis-length**.
5. The number of divisions (tick-marks) along the axis is set by **num-divisions**.
6. If **show-first** is true (**1**), the value at the start of the line is plotted.
7. If **zero-as-1** is true (**1**), a value of zero is plotted as 1 (used with sequence numbering).

```
<!ELEMENT Axis-widget (Coord,(X-axis|Y-axis))>
```

The following excerpt controls the axis displayed below.

Example 71:

```
<Axis-widget id="AXW4" caption="Distance" capt-font="FNT2"
  ascending="1" topright="0" axis-length="4.5in" start="0" end="100"
  num-divisions="5" show-first="1" zero-as-1="0">
  <Coord hloc="0.5in" vloc="2.75in">
    <X-axis use-interval="0" tick-length="0.05in" tick-width="0.01in"
      tick-color="navy" axis-font="FNT3">
      <Numbering use-numbering="1" type="REAL" a="-5.0867" b="0.0"
        dec-places="1">
      </X-axis>
    </Axis-widget>
```



Because **num-divisions** is defined (**= "5"**) in the **Axis** element, **use-interval** is set to false (**0**) and no **interval-value** is specified for the **X-axis** element. The range of the axis is defined as **0** to **100**, and a linear transformation is applied by the **Numbering** element (**n=0.0-5.0867i**). This means that the start of the line is labeled as **0.0** and the end (at value **100**) is labeled as **-508.7**, with increments of **-101.7**.

2.5.11.3 File Contents Displays

The BSML DTD defines widgets that support the display of three types of file:

1. Text files, containing simple unformatted (ASCII) text
2. Graphic image files, containing a **gif**, **jpeg** or **bmp** image
3. Table files, containing tabular data in one of the permitted formats

2.5.11.3.1 File-widget

A **File-widget** is used to display either a text file or a graphic image file. The display will be printed on the page if **own-window** is not true (**1**). (If the file is displayed in its own window, the **caption** is used as the window caption.) Regardless of the state of **own-window**, the display may be made in a separate window by using a **link** element to point to the widget (**#id**).

```
<!ELEMENT File-widget (Coord?,Border?, Image-map?, Object?,
                        Resource*, %links;) >
```

The type of file is specified in the **type** attribute. If the value is "**text**", the file is treated as a text file; otherwise it is treated as a graphics file, which must be a Windows bitmap (**bmp**), GIF, or JPEG format file. The file or url containing the data is specified in **source**. The location of the file display is controlled by the **Coord** element, which is not required if **own-window="1"**.

If the file is a text file, the **text-font** specifies the font that is used to display the text. The text is expected to be simple ASCII text with lines ended by carriage return/line feed characters. The display area is set by the **height** and **width** attributes. Only text that fits in this area is displayed. The following example defines the text display shown below.

Example 72:

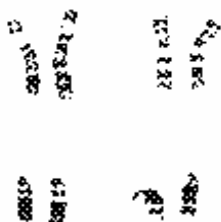
```
<File-widget id="FIL1" own-window="1" caption="Base Proportions"
  text-font="FNT4" type="text" source="CAMPBP.TXT"
  width="4in" height="2.5in">
  <Coord hloc="2.0in" vloc="1.0in">
</File-widget>
```

Pos	Raw Frequencies					Proportions			
	A	C	G	T	Sum	A	C	G	T
-15	1	1	7	2	11	0.091	0.091	0.636	0.182
-14	1	2	2	7	12	0.083	0.167	0.167	0.583
-13	3	3	2	4	12	0.250	0.250	0.167	0.333
-12	6	2	2	2	12	0.500	0.167	0.167	0.167
-11	6	2	2	3	13	0.462	0.154	0.154	0.231
-10	6	1	5	1	13	0.462	0.077	0.385	0.077
-9	6	3	1	5	15	0.400	0.200	0.067	0.333
-8	6	2	3	4	15	0.400	0.133	0.200	0.267
-7	6	2	3	4	15	0.400	0.133	0.200	0.267

If the file is a graphics file, the size of the graphic image is used to set the display area if **height** and **width** are set to **0** or are undefined. If positive values are used for the **height** and **width**, the graphic image is stretched to fit the display area. The following defines a simple graphic file display.

Example 73:

```
<File-widget type="bmp" source="CHROMOS.BMP">
  <Coord hloc="1.0in" vloc="1.0in">
</File-widget>
```



2.5.11.3.1 Image Maps

A **File-widget** for a graphic image file may also have an **Image-map** element associated with it. The **Image-map** definition follows the html 4.0 standard for client-side image maps and is used to define rectangular areas of the graphic display so that they may be used to select areas of the image. The **Image-map** may contain any number of **Area** elements, each of which defines a selection region (only rectangular areas may be defined in the current implementation, i.e., the **shape** attribute is ignored).

```
<!ELEMENT Image-map (Area)+ >
```

```
<!ELEMENT Area (ATTRIBUTE*,RESOURCE*,%LINKS;) >
```

The following example defines selection regions for the graphic image defined in **humkaryo.bml**.

Example 74:

```
<File-widget display="1" type="bmp" source="humkaryo.bmp">
  <Coord hloc="0.2in" vloc="0.5in">
    <Image-map name="human karyotypes">
      <Area shape="rect" alt="Chromosome 1" coords="66px,14px,143px,93px"
        href="http://cedar.genetics.soton.ac.uk/pub/chrom1/gene.html">
      <Area shape="rect" alt="Chromosome 2" coords="162px,5px,214px,89px"
        href="http://cedar.genetics.soton.ac.uk/pub/chrom2/gene.html">
      ...
    </Image-map>
  </File-widget>
```

2.5.11.3.2 Table-widget

A **Table-widget** is used to display tabular data, either on the page or in its own window. If **own-window="1"**, only a popup display is available. If **own-in-grid="1"**, the popup is displayed in a spreadsheet-like display. Otherwise, the table is displayed in an html-browser-type table.

The source of the data is specified by a **tableid** reference to a table defined in the **Tables** section of the **Definitions**. If **rownums** is undefined, all rows are included in the table. Otherwise, the comma-separated list of row numbers is displayed.

```
<!ELEMENT Table-widget (Coord?,Border?, Table-column+,
    Object?, Resource*, %links;) >
```

The display of the columns of the table may be controlled by attaching **Table-column** elements to the **Table-widget**.

```
<!ELEMENT Table-column EMPTY >
```

Example 75:

```
<Definitions>
...
<Table-import id="TIM1" title="Redundancy" format="SDF"
  column-numbers="1,7,8,9,10,11"
  column-headers="position,p(a),p(c),p(g),p(t),redund"
  display-widths="0.5in,0.5in,0.5in,0.5in,0.5in,0.7in"
  alignments="R,R,R,R,R,R,R" font="FNT2" add-row-numbers="0"
  locked-columns="1" source="campredu.sdf">
</Table-import>
</Definitions>
<Display>
...
<Table-widget tableid="TIM1" title="Redundancy" own-window="1">
</Table-widget>
</Display>
```

Redundancy					
position	p(a)	p(c)	p(g)	p(t)	redund
-10	0.462	0.077	0.385	0.077	19.48
-9	0.400	0.200	0.067	0.333	10.99
-8	0.400	0.133	0.200	0.267	5.57
-7	0.400	0.133	0.200	0.267	5.57
-6	0.412	0.176	0.176	0.235	4.98
-5	0.333	0.278	0.111	0.278	4.69
-4	0.158	0.263	0.263	0.316	2.10
-3	0.143	0.238	0.190	0.429	6.41
-2	0.455	0.227	0.136	0.182	7.98
-1	0.500	0.091	0.318	0.091	17.45
1	0.318	0.136	0.136	0.409	8.20
2	0.364	0.000	0.182	0.455	25.29

2.5.11.4 Explanatory Keys

The BSML DTD defines three widgets that present explanatory keys.

2.5.11.4.1 Text-key-widget

A **Text-key-widget** is used to display textual explanations for short textual captions or abbreviations.

```
<!ELEMENT Text-key-widget (Coord, Border?, Text-key-item+,
                           Resource*, %links;) >

<!ELEMENT Text-key-item EMPTY >
```

Example 76:

```
<Text-key-widget key-width="0.2in" item-gap="0.1in">
  <Coord hloc="5.3in" vloc="0.2in"/>
  <Text-key-item key="P" explanation="Plus strand"/>
  <Text-key-item key="M" explanation="Minus strand"/>
</Text-key-widget>
```

P	Plus strand
M	Minus strand

2.5.11.4.2 Fill-key-widget

A **Fill-key-widget** is used to display textual explanations for fill patterns used with **Interval-display-widget** elements.

```
<!ELEMENT Fill-key-widget (Coord, Border?, Resource*, %links;) >
```

Example 77:

```
<Fill-key-widget font="FNT1" pattern-width="0.5in"
  pattern-height="0.25in" line-width="0.02in" pattern-gap="0.1in"
  patterns="clear,diagcross" fore-colors="black,black"
  back-colors="white,white" explanations="intron,exon">
  <Coord hloc="5.35in" vloc="2.7in"/>
</Fill-key-widget>
```

	intron
	exon

2.5.11.4.3 Symbol-key-widget

A **Symbol-key-widget** is used to display textual explanations for symbols used with **Point-display-widget** and **Point-group-display-widget** elements.

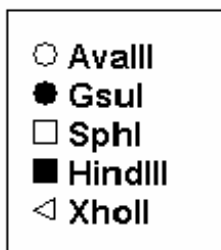
```
<!ELEMENT Symbol-key-widget (Coord,Border?, Resource*, %links;) >
```

Example 78:

```

<Symbol-key-widget do-substitute="1" font="FNT4" size="0.1in"
  line-width="0.01in" color="blue" open-circle="AvaIII"
  closed-circle="GsuI"
  open-square="SphI" closed-square="HindIII" triangle="XhoII">
  <Coord hloc="5.0in" vloc="1.0in"/>
  <Border/>
</Symbol-key-widget>

```



2.5.11.5 Unaligned Data Chart

An unaligned chart is displayed as a conventional "stand-alone" data plot (not along a sequence line in the manner of an **aligned chart** display). The **Chart-widget** may obtain its data from one of three sources:

1. Data defined in the element itself
2. Data obtained from a **Table** defined in the **Definitions** section
3. Data obtained from an external data file

```

<!ELEMENT Chart-widget (Coord?,X-chart-axis?,Y-chart-axis?,
  Chart-page?,Chart-screen-display?,Chart-print-display?,
  Chart-data*,Object?, Resource*, %links;) >

<!ELEMENT (X-chart-axis|Y-chart-axis) EMPTY >

<!ELEMENT Chart-page EMPTY >

<!ELEMENT (Chart-screen-display|Chart-print-display) EMPTY >

<!ELEMENT Chart-data EMPTY >

```

The following example plots the chart shown below, using data obtained from file **campredu.sdf**.

Example 79:

```

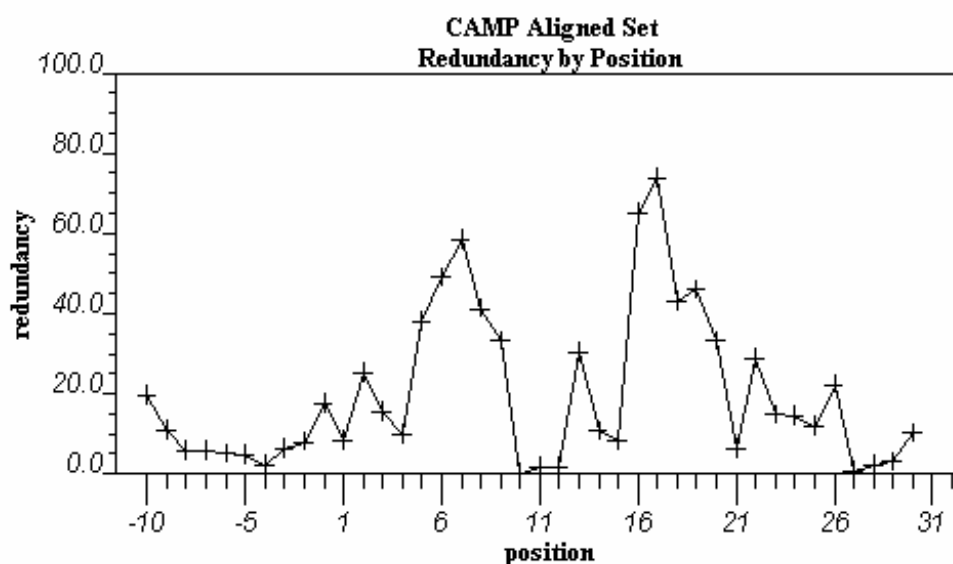
<Chart-widget onclick="LNK1" own-window="0" width="5in" length="3in"
  chart-type="xy" format="sdf" source="campredu.sdf" bycolumn="1"
  title-1="CAMP Aligned Set" title-2="Redundancy by Position"
  x-title="position" y-title="redundancy" x-no-zero="1">
  <Coord hloc="0.8in" vloc="0.7in">
  <X-chart-axis scale-automatic="1" display-mode="decimal-places"
    dec-places="0">
  <Y-chart-axis scale-automatic="0" lower-limit="0.0"

```

```

upper-limit="100.0" display-mode="best-fit" dec-places="0">
<Chart-page left="1.0" top="1.0" width="5.0" length="4.0"
portrait="1">
<Chart-screen-display monochrome="0"plot-thickness="0.01in"
frame-thickness="0.01in" plot-color="black" frame-color="black"
plot-background-color="white" frame-background-color="white"
symbol-size="0.1in">
<Chart-print-display monochrome="0"plot-thickness="0.01in"
frame-thickness="0.01in" plot-color="black" frame-color="black"
plot-background-color="white" frame-background-color="white"
symbol-size="0.1in">
<Chart-data setnum="0" colrownum="1">
<Chart-data setnum="1" line-color="red" symbol="plus" colrownum="11">
  <Link id="LNK1" title="View Data" href="#TIM1">
</Chart-widget>

```



2.5.11.6 Sequence-related Data Displays

Two elements display sequence data, but are "stand alone" in the sense that they do not require references to **Sequence** elements.

2.5.11.6.1 Gel-widget

A **Gel-widget** plots a simulated electrophoresis gel, using data points supplied as either fragment sizes or cut positions. If cut positions are supplied, fragment lengths are computed using the length and shape (linear, circular) of the sequence.

```

<!ELEMENT Gel-widget (Coord,Border?,Digest-set*,Object?,
Resource*, %links;) >

```

Note that **Digest-set** and **Restriction-enzyme** elements are described above, in the section on **Definitions**. The content models are repeated here for convenience.

```

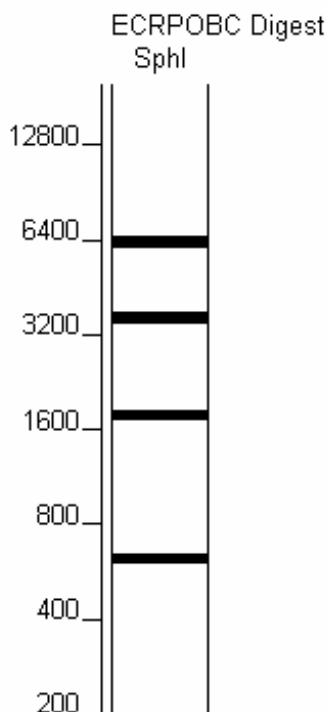
<!ELEMENT Digest-set (Restriction-enzyme)* >

```

<!ELEMENT Restriction-enzyme EMPTY >

Example 80:

```
<Gel-widget title="ECRPOBC Digest" title-font="FNT3" min-length="200"
max-length="20000" min-label="200" scale-factor="1.0"
label-decimals="1" label-ratio="2.0" label-font="FNT3"
gel-length="0@idh:VEW0" lane-width="0.5in" lane-gap="0.05in"
lane-thick="0.01in" lane-color="black" band-prop="1"
band-thick="0.1in" band-low="0.05in" band-color="navy"
confusion="0,01in" confusion-color="red" plot-big="1" use-key="0">
  <Coord hloc="2.0in" vloc="0.2in">
    <Digest-set title="SphI" seqref="SEQ1">
      <Restriction-enzyme title="SphI" source="ECRPOBC"
        recognition-seq="GCATGC" offset="5" overhang="-4"
        sites="628,6926,8710">
    </Restriction-enzyme>
  </Digest-set>
</Gel-widget>
```



2.5.11.6.2 Seq-data-widget

A **Seq-data-widget** displays bases or residues for sequence data obtained from one of three sources:

1. A **Sequence** element defined in the **Definitions** section
2. A **Seq-data** element defined for the **Seq-data-widget**
3. An external file of sequence data referred to by the **source** attribute


```
<!ELEMENT Seq-data-widget (Coord,Border?,Seq-data?,Object?,
                           Resource*, %links;)>
```

Example 81:

```
<Seq-data-widget startpos="1" length="140"
  caption="translation listing" seq-font="FNT3" caption-font="FNT3"
  bases-per-line="50" number-bases="1" space-blocks="1" is-protein="1">
  <Coord hloc="0.5in" vloc="3in">
    <Seq-data>
      mdkvlnreeslqlmdllglersawgniplmrkaylkkckefhpdkggdeekmkkmntlykkmedgv
      kyahqpdfggfwdateiptygtdeweqwwnafneenlfcseempssddeatadsqhstppkkkrkv
    </Seq-data>
  </Coord>
</Seq-data-widget>
```

```
translation listing
1 mdkvlnrees lqlmdllgle rsawgniplm rkaylkkcke fhpdkggdee
51 kkkmntlyk kmedgvkyah qpdfggfwda teiptygtde weqwwnafne
101 enlfcseemp ssddeatads qhstppkkkr kv
```

2.5.12 Sequence View Display Widgets

A sequence **View** refers to one of the **Sequence** elements defined in the **Definitions** section of a BSML document. A **View** plots a sequence line as either a linear or circular representation of the sequence, and the line may be single- or double-stranded.

```
<!ELEMENT View (Numbering?, Margin?,%seqdisplaywidgets;,
               Object?, Resource*, %links;)>
```

Collectively, the elements forming the content model of a **View** element are defined by the **seqdisplaywidgets** entity as:

```
<!ENTITY % seqdisplaywidgets "(Point-widget|Point-group-widget|
Interval-widget|Blowup-widget| View-axis-widget|View-line-widget|
Aligned-chart-widget)*">
```

2.5.12.1 View-line-widget

The **View-line-widget** controls the display of the sequence line independently of the **Sequence** element that is the reference element for the **View**.

```
<!ELEMENT View-line-widget EMPTY >
```

Example 82:

```
<View-line-widget id="VLN1" shape="circular" hcenter="4.31in"
  vcenter="1.95in" circular-diam="2.68in" strands="one"
  plus-color="blue" minus-color="black"/>
```

2.5.12.2 View-axis-widget

The **View-axis-widget** controls the location and presentation of numbering information on the sequence line.

```
<!ELEMENT View-axis-widget EMPTY >
```

Example 83:

```
<View-axis-widget id="AXS1" display="1" linear-at-side="0"
  circular-rotate="0" base-at-top="1" use-interval="0"
  interval-value="100"
  axis-strand="minus" tick-color="blue" axis-font="FNT2"/>
```

2.5.12.3 Point-widget

A **Point-widget** displays a feature at a single location on the sequence line and may refer to an underlying **Feature** through its **featureref** attribute. If **featureref** is defined, selecting this **Point-widget** accesses information from the associated **Feature**.

```
<!ELEMENT Point-widget (Symbol?, Object?, Resource*, %links;) >
```

Example 84:

```
<Point-widget caption="polyA_site" class="POLYA_SITE"
  comment="constructed display feature" featureref="FTR11"
  id="DEFFTR9" position="1946" title="polyA_site">
</Point-widget>
```

2.5.12.4 Point-group-widget

A **Point-group-widget** defines a set of points on the sequence line that may refer to underlying **Feature** elements. The points are plotted collectively to avoid overlapping text. The points are also formatted collectively in terms of font, color, spacing, etc.

```
<!ELEMENT Point-group-widget (Quantifier?, Member*, Object?,
  Resource*, %links;) >
```

The elements comprising the **Point-group-widget** are defined by **Member** elements.

```
<!ELEMENT Member EMPTY >
```

Example 85:

```
<Point-group-widget on-strand="source" auto-plot="0" one-vert-col="0"
  show-text="1" show-position="0" line-gap="0.00in"
  text-gap="0.10in" line-width="0.01in"
  line-color="black" line-length="0.1in">
  <Member caption="AvaIII" position="131" plus="1"/>
  <Member caption="HindIII" position="5172" plus="1"/>
</Point-group-widget>
```

2.5.12.5 Interval-widget

An **Interval-widget** displays a feature on a region of the sequence line and may refer to an underlying **Feature** through its **featureref** attribute. If **featureref** is defined, selecting this **Interval-widget** accesses information from the associated **Feature**.

```
<!ELEMENT Interval-widget (Object?, Resource*, %links;)>
```

Example 86:

```
<Interval-widget startpos="230" endpos="400" on-sequence="center"
  gap-in-seq-line="1" show-positions="no" line-width="0.02in"
  fill-pattern="clear" border-color="blue" fill-fg-color="blue"
  fill-bg-color="aqua" offset-from-seq="0.05in"
  arrow-start-len="0.06in" arrow-end-len="0.20in"
  arrow-width="0.20in" bracket-text-gap="0.00in"
  bracket-line-gap="0.00in" bracket-color="black"
  plus-bracket="none" minus-bracket="none"
  center-text="Interval (m)" center-font="FNT6"
  start-type="arrow" end-type="none">
</Interval-widget>
```

2.5.12.6 Blowup-widget

A **Blowup-widget** displays text, typically bases or residues, aligned to a particular place on the **View** line.

```
<!ELEMENT Blowup-widget (Object?, Resource*, %links;) >
```

Example 87:

```
<Blowup-widget display="1" startpos="490" endpos="500" on-plus="0"
  line-width="0.02in" fill-pattern="clear" border-color="red"
  fill-fg-color="blue" fill-bg-color="yellow"
  bracket-text-gap="0.00in"
  bracket-line-gap="0.00in" bracket-line-len="0.60in"
  bracket-line-wid="0.02in" bracket-color="red"
  cut-thickness="0.05in"
  plus-text="blow^up" plus-font="FNT10" minus-text="pu^wolb">
</Blowup-widget>
```

2.5.12.7 Aligned-chart-widget

```
<!ELEMENT Aligned-chart-widget (Chart, Quantifier?, Object?,
                                Resource*, %links;)>
```

An **Aligned-chart-widget** obtains its data through a **Chart** element, which selects a row or column from a data table identified by url or **id** reference.

```
<!ELEMENT Chart (Border?)>
```

Example 88:

```
<Aligned-chart-widget onclick="LNK91" startpos="1" endpos="708"
  min-offset="0.50in" max-offset="1.25in" min-data="-2" max-data="2">
```

```

dec-places="1" show-y-axis="1" histogram="0" line-width="0.01in"
line-color="black" caption="hydrophobicity" caption-font="FNT3"
field-number="1" data-font="FNT2" id="ALC0" full-only="0" on-plus="1"
min-points="0">
<Chart source="ltaghydr.dat" colrownum="3">
  <Border line-width="0.01in" line-color="red" gutter="0.05in"/>
</Chart>
<Quantifier type="saturation" min-val="20.0" max-val="30.0"
  val-name="red" base="0" slope="255"/>
<Link id="LNK91" href="#TIM1" title="hydrophobicity table"
  show="embed" actuate="user"/>
</Aligned-chart-widget>

```

2.5.12.8 View-master-widget

A **View-master-widget** refers to a set of **View** elements through its **views** attribute. A navigation widget is displayed that allows movement back and forth through the set of **Views**.

```
<!ELEMENT View-master-widget (Coord, Object?, Resource*, %links;) >
```

Example 89:

```

<View-master-widget
  views="VEW1 VEW2 VEW3 VEW4 VEW5 VEW6 VEW7 VEW8 VEW9 VEW10"
  width="1.4in" height="0.4in" outerwidth="0.02in" outerborder="navy"
  outerfill="silver" innerwidth="0.02in" innerborder="navy"
  innerfill="aqua" shapeborder="maroon" shapefill="green"
  disabledborder="gray" disabledfill="gray" shape="vertical"
  hcenter="2.0in" vcenter="2.5in"
  hoffset="2.0in" voffset="0in" numviews="3" linear-length="4.0in"
  circular-diam="2.5in">
  <Coord hloc="0.0in" vloc="0.0in"/>
</View-master-widget>

```

2.5.12.9 Sequence-viewer

A **Sequence-viewer** widget is a non-visual element that describes the conditions under which a sequence should be viewed in the sequence viewer window of the Genomic XML Viewer.

```
<!ELEMENT Sequence-viewer (Class-key*, Object?, Resource*, %links;) >
```

```
<!ELEMENT Class-key EMPTY>
```

Example 90:

```

<Sequence-viewer id="SWD1" title="TEST" class="TEST" refseq="SEQ1"
  complete-seq="0" view-start="1" view-end="2000" maximize-window="1"
  spacing="5" fit-to-window="0" bases-per-line="500"
  bases-per-window="200" int-pos="5" interval-height-min="0"
  interval-height-max="10" font-pos="5" font-size-min="6"
  font-size-max="15" show-sites="1" sites-as-ints="0"
  sites-not-full="0" sites-by-pct="0" sites-percent="0"

```

```

interval-pos="3" interval-overlap="0" interval-separate="0"
interval-by-width="0" int-pixels="10" int-as-point="1"
plus-strand="1" minus-strand="1" frame="0"
show-point-titles="1" show-interval-titles="1"
interval-titles-above="1" interval-title-once="1" title-length="12"
show-key="1" feature-histogram="1" feature-position="3"
density-per-window="1" density-method="2" density-threshold="10"
nbins="25" pixels-per-countX1000="1000" max-histo-height="25"
auto-fit-histogram="0">
<Class-key key-title="SOURCE" key-class="SOURCE" border-color="blue"
  fill-color="yellow" row="1" enabled="1"/>
<Class-key key-title="POLYA_SIGNAL" key-class="POLYA_SIGNAL"
  border-color="blue" fill-color="aqua" row="2" enabled="1"/>
<Class-key key-title="Gene" key-class="GENE" border-color="blue"
  fill-color="lime" row="3" enabled="1"/>
</Sequence-viewer>

```

2.5.12.10 Feature-histogram-widget

A **Feature-histogram-widget** displays a histogram of the distribution of features on a View.

```
<!ELEMENT Feature-histogram-widget (Coord,Object?, Resource*, %links;) >
```

Example 91:

```

<Feature-histogram-widget viewref="VEW1" caption="Feature Histogram"
  features-histogram="1" feature-position="0" nbins="20"
  height-per-bin="0.1in" line-width="0.01in" bin-width="0.1in"
  line-color="black" fill-bins="1" maximum-height="1in" auto-fit="1"
  capt-font="FNT1" numb-font="FNT2">
  <Coord hloc="0.50in" vloc="0.67in"/>
</Feature-histogram-widget>

```

2.5.13 Display Set Widgets

Several display widgets are related to sets of objects that are viewed either collectively or in relation to one another.

2.5.13.1 Simple-set-widget

A **Simple-set-widget** displays the member of one or more **Set** elements.

```
<!ELEMENT Simple-set-widget (Coord,Border?, Set*, Object?,
  Resource*, %links;)
```

Example 92:

```

<Simple-set-widget id="SEQSET" own-window="0" setids="SET1"
  caption="Simple Set" capt-font="FNT1" list-font="FNT2" align="left"
  line-link="1" line-width="0.02in" line-color="blue" depth="3"
  indent="0.2in">
  <Coord hloc="1.0in" vloc="1.0in"/>

```

```
</Simple-set-widget>
```

2.5.13.2 Tree-set-widget

A **Tree-set-widget** represents a hierarchy displayed as a tree (directed graph). The basic display consists of titles obtained from a list of **ids**; the length of a branch specified in the **value** attribute is set relative to **branchunit** and **branchbase** attributes.

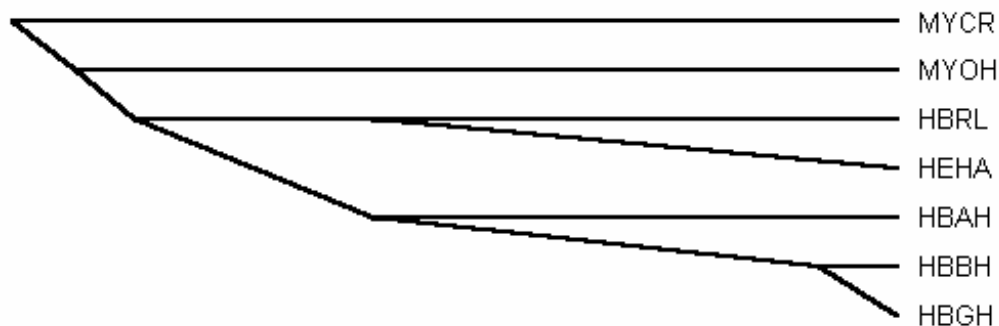
```
<!ELEMENT Tree-set-widget (Coord,Border?,Tree-node*,Object?,
                           Resource*, %links;) >
```

```
<!ELEMENT Tree-node EMPTY>
```

Example 93:

```
<Tree-set-widget id="TRE1"
  caption="Globin Gene Multiple Alignment (click to view)"
  capt-font="FNT1" branch-cap-font="FNT3" show-leaf-values="1"
  link-straight="0" node-font="FNT3" line-width="0.02in"
  line-color="maroon" branchlen="0.0in" text-gap="0.1in">
  <Coord hloc="0.5in" vloc="0.5in">
  <Tree-node id="TND1" hloc="0.5in" vloc="1.0in">
  <Tree-node id="TND2" hloc="5.00in" vloc="1.00in" setid="SEQ7"
    parentid="TND1">
  <Tree-node id="TND3" hloc="0.82in" vloc="1.25in" parentid="TND1">
  <Tree-node id="TND4" hloc="5.00in" vloc="1.25in" setid="SEQ6"
    parentid="TND3">
  <Tree-node id="TND5" hloc="1.13in" vloc="1.50in" parentid="TND3">
  <Tree-node id="TND6" hloc="2.33in" vloc="1.50in" parentid="TND5">
  <Tree-node id="TND7" hloc="5.00in" vloc="1.50in" setid="SEQ5"
    parentid="TND6">
  <Tree-node id="TND8" hloc="5.00in" vloc="1.75in" setid="SEQ4"
    parentid="TND6">
  <Tree-node id="TND9" hloc="2.33in" vloc="2.00in" parentid="TND5">
    <Tree-node id="TND10" hloc="5.00in" vloc="2.00" setid="SEQ3"
      parentid="TND9">
  <Tree-node id="TND11" hloc="4.60in" vloc="2.25in" parentid="TND9">
  <Tree-node id="TND12" hloc="5.00in" vloc="2.25in" setid="SEQ2"
    parentid="TND11">
  <Tree-node id="TND13" hloc="5.00in" vloc="2.50in" setid="SEQ1"
    parentid="TND11">
</Tree-set-widget>
```

Globin Gene Multiple Alignment



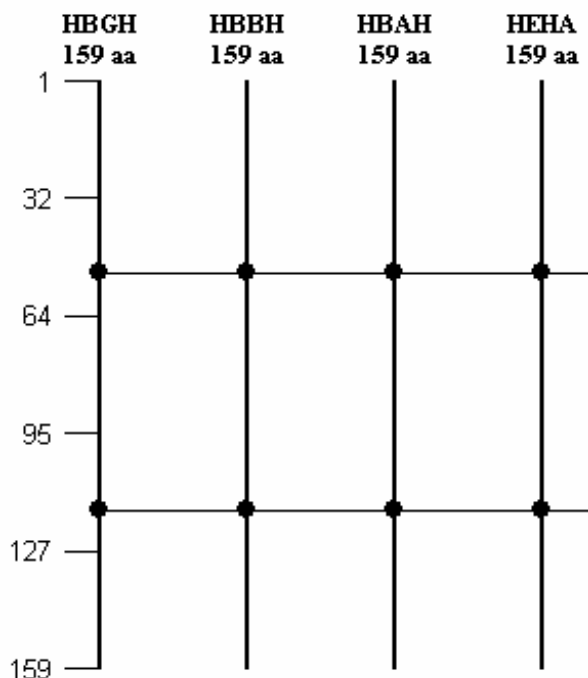
2.5.13.3 Alignment-point-sets-widget

An **Alignment-point-sets-widget** obtains alignment points for a collection of **Alignment-point-set** elements, each of which indicates a position on a sequence. The display involves linking the points.

```
<!ELEMENT Alignment-point-sets-widget
  (Symbol?, Alignment-point-set*, Object?, Resource*, %links;) >
```

Example 94:

```
<Alignment-point-sets-widget line-width="0.01in" line-color="blue">
  <Symbol shape="closedcircle" diam="0.1in" color="blue">
    <Alignment-point-set seqids="V3 V4 V5 V6 V7 V8 V9"
      featids="52 52 52 52 52 52 52"
      numseg="1" starts="1,1,1,1,1,1,1" seglens="159" >
    </Alignment-point-set>
  </Alignment-point-sets-widget>
```



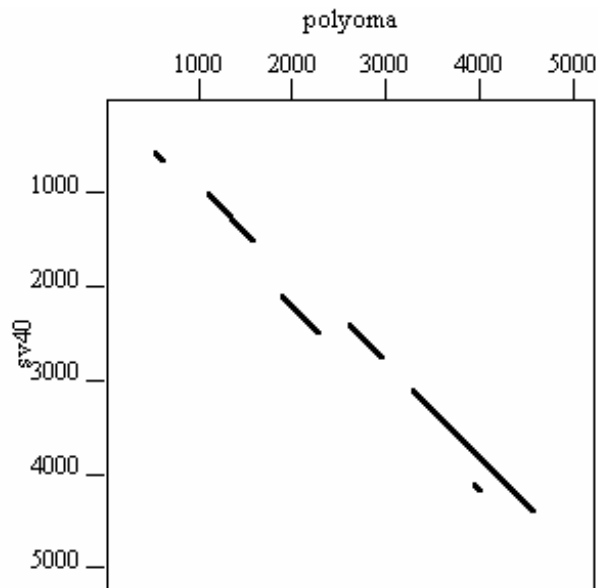
2.5.14 Dot-plot-widget

A **Dot-plot-widget** is used to represent the relationship between two sequences.

```
<!ELEMENT Dot-plot-widget (Coord,
    X-axis?,Y-axis?,Quantifier?,Object?, Resource*, %links;) >
```

Example 95:

```
<Dot-plot-widget seqpair="SPA1" width="2.5in" height="2.5in"
  use-x-axis="1" use-y-axis="1" y-proportional="1" y-on-top="1"
  line-width="0.02in" line-color="blue" border-width="0.01in"
  border-color="red">
  <Coord hloc="0.7in" vloc="0.5in"/>
  <X-axis use-interval="1" interval-value="1000" tick-length="0.1in"
    tick-width="0.01in" tick-color="black" axis-font="FNT2">
  </X-axis>
  <Y-axis use-interval="1" interval-value="1000" tick-length="0.1in"
    tick-width="0.01in" tick-color="black" axis-font="FNT2">
  </Y-axis>
  <Quantifier type="color" min-val="0.0" max-val="30.0" red-base="255"
    red-slope="-255" blue-base="0" blue-slope="255" green-base="0"
    green-slope="0"/>
</Dot-plot-widget>
```

2.5.15 Network-display-widget [3]

A **Network-display-widget** provides visualization elements for a **Network** element.

```
<!ELEMENT Network-display-widget
  (Coord?, Node-display*, Arc-display*, Resource*, %links; ) > [3]

<!ELEMENT Node-display EMPTY > [3]

<!ELEMENT Arc-display EMPTY > [3]
```

3. BSML DTD Reference

This section of the manual provides the detailed presentation of each element in the DTD and a discussion, as warranted, of each attribute. First, the entity definitions used in BSML are presented.

3.1 Parameter Entities

Entities for Text Characters

These entities are used to enter symbols that may not appear in parsed XML text.

<u>Entities</u>	<u>Symbol</u>
<code><!ENTITY amp CDATA "&#38;#38;"></code>	<code>&</code>
<code><!ENTITY lt CDATA "&#38;#60;"></code>	<code><</code>
<code><!ENTITY gt CDATA "&#62;"></code>	<code>></code>
<code><!ENTITY quot CDATA "&#34;"></code>	<code>"</code>
<code><!ENTITY apos CDATA "&#39;"></code>	<code>'</code>

Entities for Attribute Constants

The following entities specify the intended values and usage for a variety of constants that can define attribute values

<u>Entities</u>	<u>Explanation</u>
<code><!ENTITY % yesorno "CDATA"></code>	boolean as 1=true 0=false
<code><!ENTITY % integer "CDATA"></code>	positive or negative int
<code><!ENTITY % real "CDATA"></code>	positive or negative real
<code><!ENTITY % length "CDATA"></code>	length units (see 1.1)
<code><!ENTITY % linewidth "CDATA"></code>	line width (see 1.2)
<code><!ENTITY % linestyle "(solid dash dot none)"></code>	style of line
<code><!ENTITY % displayshape "(ellipse rectangle roundrectangle)"></code>	
<code><!ENTITY % displaystate "(on off gray)"></code>	
-- connector style for networks [3] --	
<code><!ENTITY % connectortype "(none ball arrow)"></code>	[3]
<code><!ENTITY % textposition "(above below left right)"></code>	[3]

The following entities were adapted from html 4.0:

<code><!ENTITY % color "CDATA"></code>	name, RGB value
<code><!ENTITY % contenttype "CDATA"></code>	media type,RFC2045
<code><!ENTITY % character "CDATA"></code>	one character,ISO10646
<code><!ENTITY % mediadesc "CDATA"></code>	single or comma-separated list of media descriptors
<code><!ENTITY % url "CDATA" ></code>	Uniform Res Locator,RFC1808,RFC1738
<code><!ENTITY % datetime "CDATA"></code>	date,time ISO 8601 format, e.g., 2001-08-01T13:04:00
<code><!ENTITY % date "CDATA"></code>	ISO 8601 format, e.g., 2001-01-31
<code><!ENTITY % script "CDATA"></code>	script expression
<code><!ENTITY % text "CDATA"></code>	render as CDATA

```
<!ENTITY % shape "(rect|circle|poly|default)"> image maps
<!ENTITY % coords "CDATA"> comma sep list of numbers
<!ENTITY % frametarget "CDATA"> render in this frame
```

Notes

1. A network url MUST include the prefix http://
2. A local file url MAY include the prefix file://

Datatype Entity

This entity is used in the definition of datatypes for the **Attribute** element's **datatype** attribute.

```
<!ENTITY % datatype "(integer|float|character)"
```

Location Entity

This entity includes either type of location (site or interval) in a content model. See Location elements.

```
<!ENTITY % location "(Site-loc|Interval-loc)">
```

Dbsource Entity

A database source defines a controlled vocabulary for identifying sources (e.g., GenBank).

```
<!ENTITY % dbsource "CDATA" >
```

Segment-opts Entity

This entity defines the enumerated values allowed for **Sequence** segments.

```
<!ENTITY % segment-opts
"(chromosome|chromosomal-contig|clone|sequence-contig|sequence|other)">
```

Repr-opts Entity

This entity defines sequence representation types (as described by NCBI). See the **Sequence** element.

```
<!ENTITY % repr-opts "(not-set|virtual|raw|segmented|constructed|
reference|consensus|map|derived|other)" >
```

Timeunits Entity

This entity is used in research representation.

```
<!ENTITY %  
"(msec|seconds|minutes|hours|days|weeks|months|year)">
```

timeunits

Links Entity

The links entity defines the link elements (including **Attribute-list** elements) as any number of occurrences of any link element. See Linking elements. Note that the **Cross-reference** element was added to this list for BSML 3.0.

```
<!ENTITY % links "(Attribute-list|Cross-reference|Link|Extended-  
link|Group-link)*">
```

Setwidgets Entity

This entity describes display widgets related to sets and is used in the **Page** content model.

```
<!ENTITY % setwidgets "(Simple-set-widget|Tree-set-widget|  
Alignment-point-sets-widget|Dot-plot-widget)" >
```

Seqdisplaywidgets Entity

This entity defines the elements that form the content model of a **View** element.

```
<!ENTITY % seqdisplaywidgets "(Point-widget|Point-group-widget|  
Interval-widget|Blowup-widget|  
View-axis-widget|View-line-widget|  
Aligned-chart-widget)*">
```

Pagedisplaywidgets Entity

This entity describes a set of elements in the **Page** element content model.

```
<!ENTITY % pagedisplaywidgets "(Popup-text-widget|Sequence-viewer|  
Feature-histogram-widget|Caption-widget|Line-pointer-widget|  
Shape-widget|Table-widget|File-widget|Gel-widget|Axis-widget|  
Text-key-widget|Fill-key-widget|Symbol-key-widget|  
Seq-data-widget|Chart-widget|View-master-widget|  
Network-display-widget)">
```

3.2 Basic Attributes and Elements

Core Attributes

The core attributes consist of the **mainattrs** and **events** entities.

```
<!ENTITY % attrs "%mainattrs; %events;" >
```

Notes

% **attrs** is added as an attribute group for all major elements.

Mainattrs Entity

The entity provides attributes that give an element general characteristics used for collectively formatting groups of elements. Note that **editstatus** replaced the BSML 2.2 **style** attribute in BSML 3.0.

Attributes

```
<!ENTITY % mainattrs
" id          ID          #IMPLIED
  class       CDATA       #IMPLIED
  style       CDATA       #IMPLIED [2.2]
  editstatus  (unchanged|added|edited|deleted) "unchanged" [3]
  title       CDATA       #IMPLIED
  comment     CDATA       #IMPLIED
  value-type  CDATA       #IMPLIED
  value       CDATA       #IMPLIED
  selectable  %yesorno;   #IMPLIED
  selected    %yesorno;   #IMPLIED
  display     %yesorno;   #IMPLIED
  readonly    %yesorno;   #IMPLIED">
```

Notes*

<u>Attribute</u>	<u>Explanation</u>
id	document-wide unique id
class	space-separated class list
style	inline style specification
editstatus	indicator of a change to an element
title	displayable title
comment	non-displayed comment
value-type	optional value description
value	optional data value(s)
selectable	selectable by mouse click
selected	element is selected
display	show element if page plotted
readonly	allow editing if not readonly

- * Value-type and value are used, as appropriate, to supply data or other numeric values associated with an element.
- * Value may be a single entry or a comma-separated list.

Events Entity

This entity defines actions to be taken in response to **mouse (pointer)** and **keyboard** events associated with an element.

Attributes

```
<!ENTITY % events
"onclick      %script;    #IMPLIED
ondblclick    %script;    #IMPLIED
onmousedown   %script;    #IMPLIED
onmouseup     %script;    #IMPLIED
onmouseover   %script;    #IMPLIED
onmousemove   %script;    #IMPLIED
onmouseout    %script;    #IMPLIED
onkeypress    %script;    #IMPLIED
onkeydown     %script;    #IMPLIED
onkeyup       %script;    #IMPLIED">
```

Notes*

Attributes

onclick

ondblclick

onmousedown

onmouseup

onmouseover

onmousemove

onmouseout

onkeypress

onkeydown

onkeyup

Explanation

pointer was clicked

pointer was double clicked

pointer was pressed down

pointer was released

pointer was moved onto

pointer was moved within

pointer was moved away

key was pressed and released

key was pressed down

key was released

script is a **cdata** entity that defines the behavior to follow the event.

Direction Entity [3]

The direction entity is used by the **Network-display-widget** set of elements.

```
<!ENTITY % direction "(default|north|northeast|east|southeast|south
|southwest|west|northwest)">
```

Attribute Element

The **Attribute** element is used in the content model of many elements and defines pairs of **name** and **content** attributes. Note that the **type** attribute was added for BSML 3.0.

Content Model

```
<!ELEMENT Attribute EMPTY >
```

Attributes

```
<!ATTLIST Attribute
```

```

name      CDATA    #REQUIRED
content   CDATA    #IMPLIED
type      %datatype #IMPLIED> [3]

```

Attribute-List and Info Elements

Each of these is a list holder for a set of any number of **Attribute** elements.

Content Model

```
<!ELEMENT (Attribute-list | Info) (Attribute)* >
```

Attributes

```

<!ATTLIST (Attribute-list | Info)
  %attrs;
  idref IDREF #IMPLIED >

```

Numbering Element

The **Numbering** element specifies units of measurement for sequences and sequence-related elements and may define relative numbering basis. The **seqref** attribute added for BSML 3.0 allows explicit reference for numbering to another sequence.

Content Model

```
<!ELEMENT Numbering EMPTY >
```

Attributes

```

<!ATTLIST Numbering
  %attrs;
  use-numbering %yesorno; #IMPLIED
  type (continuous,real,enumerated,reference) #IMPLIED
  units      CDATA      #IMPLIED
  a          %real;     #IMPLIED
  b          %real;     #IMPLIED
  dec-places %integer;  #IMPLIED
  refnum     %integer;  #IMPLIED
  has-zero   %yesorno;  #IMPLIED
  ascending  %yesorno;  #IMPLIED
  names      CDATA      #IMPLIED
  from-aligns %yesorno; #IMPLIED
  aligns     IDREF      #IMPLIED
  seqref     IDREF      #IMPLIED [3]
>

```

Notes

Attribute

use-numbering if no,absolute #
 type (continuous,real,enumerated,reference)

real type

units	for display purposes--real=a N + b
a	slope
b	intercept

Explanation

dec-places %integer	
	continuous type
refnum	base 1
has-zero	else 0 reported -1
ascending	numbering order
names	comma sep enum list
	reference type
from-aligns	else from sources
aligns	
seqref	IDREF of reference sequence

Resources

BSML 3.0 documents can include elements that identify and describe associated agents or resources such as organizations, software systems, and bibliographic sources.

Resource Element

Content Model

```
<!ELEMENT Resource (Attribute | Coverage | Description | Type |
Relation | Source | Subject | Title | Contributor | Creator | Publisher
| Rights | Date | Format | Identifier | Language | Content | Version |
History | Authority | %links;)* >
```

Attributes

```
<!ATTLIST Resource
  id          ID          #IMPLIED
  title       CDATA       #IMPLIED
  url         %url;       #IMPLIED>
```

Notes

<u>Attributes</u>	<u>Explanation</u>
title	optional resource title
url	optional resource link

Postal-address Element

Content Model

```
<!ELEMENT Postal-address EMPTY >
```

Attributes

```
<!ATTLIST Postal-address
  building-address1 CDATA #IMPLIED
  building-address2 CDATA #IMPLIED
  street-address1   CDATA #IMPLIED
  street-address2   CDATA #IMPLIED
```


city	CDATA #IMPLIED
state-province	CDATA #IMPLIED
country	CDATA #IMPLIED
postal-code	CDATA #IMPLIED
utc-time-zone	CDATA #IMPLIED

Notes

Attributes

utc

Explanation

coordinate universal time - format: +01:00 =
zulu + 1 hour

Contact-info Element

Content Model

<!ELEMENT Contact-info (Postal-address?) >

Attributes

```
<!ATTLIST Contact-info
  telephone-number    CDATA #IMPLIED
  cellular-number      CDATA #IMPLIED
  fax-number          CDATA #IMPLIED
  pager-number        CDATA #IMPLIED
  department          CDATA #IMPLIED
  url                 %url; #IMPLIED>
```

Person Element

Content Model

<!ELEMENT Person (Contact-info?) >

Attributes

```
<!ATTLIST Person
  fullname            CDATA #IMPLIED
  lastname            CDATA #IMPLIED
  firstname           CDATA #IMPLIED
  middlename          CDATA #IMPLIED
  title              CDATA #IMPLIED
  department          CDATA #IMPLIED
  url                 %url; #IMPLIED>
```

Organization Element

Content Model

<!ELEMENT Organization (ContactInfo?,Person*) >

Attributes

```
<!ATTLIST Organization
```

name	CDATA	#REQUIRED
description	CDATA	#IMPLIED
department	CDATA	#IMPLIED
url	%url;	#IMPLIED>

Software-system Element

Content Model

```
<!ELEMENT Software-system (ContactInfo?, Person*, Organization*, Version*)>
```

Attributes

```
<!ATTLIST Software-system
  name          CDATA          #REQUIRED
  description    CDATA          #IMPLIED
  url            %url;          #IMPLIED>
```

Creator Element

A person, organization, or software system that is connected with a resource through authorship or ownership.

Content Model

```
<!ELEMENT Creator (Person|Organization|Software-system)+ >
```

Contributor Element

A person, organization, or software system that is connected with a resource through some type of contribution.

Content Model

```
<!ELEMENT Contributor (Person|Organization|Software-system)+ >
```

Publisher Element

Publisher of a document or other resource.

Content Model

```
<!ELEMENT Publisher (Person|Organization|Software-system)+ >
```

Title Element

Resource title.

Content Model

<!ELEMENT Title (#PCDATA) >

Type Element

Resource type.

Content Model

<!ELEMENT Type (#PCDATA) >

Subject Element

Description of subject matter.

Content Model

<!ELEMENT Subject (#PCDATA) >

Attributes

<!ATTLIST Subject		
name	CDATA	#IMPLIED
description	CDATA	#IMPLIED
url	%url;	#IMPLIED>

Relation Element

Description of relationship.

Content Model

<!ELEMENT Relation (#PCDATA) >

Attributes

<!ATTLIST Relation		
type	CDATA	#IMPLIED
url	%url;	#IMPLIED>

Format Element

Description of resource media.

Content Model

<!ELEMENT Format (#PCDATA) >

Attributes

<!ATTLIST Relation		
extent	CDATA	#IMPLIED
medium	CDATA	#IMPLIED>

Source Element

Description of the source of a resource.

Content Model

```
<!ELEMENT Source (#PCDATA) >
```

Attributes

```
<!ATTLIST Source
  name          CDATA          #IMPLIED
  description    CDATA          #IMPLIED
  url            %url;          #IMPLIED>
```

Language Element

Description of the language(s) in which the resource is available.

Content Model

```
<!ELEMENT Language (#PCDATA) >
```

Attributes

```
<!ATTLIST Language
  name          CDATA          #IMPLIED
  description    CDATA          #IMPLIED
  url            %url;          #IMPLIED>
```

Coverage Element

Description of the content area(s) covered by the resource.

Content Model

```
<!ELEMENT Coverage (#PCDATA) >
```

Attributes

```
<!ATTLIST Coverage
  name          CDATA          #IMPLIED
  description    CDATA          #IMPLIED>
```

Rights Element

Description of copyrights, trademarks, and patents that may apply to the resource.

Content Model

```
<!ELEMENT Rights (#PCDATA) >
```

Attributes

```
<!ATTLIST Rights
```

name	CDATA	#IMPLIED
description	CDATA	#IMPLIED
url	%url;	#IMPLIED>

Version Element

Description of the version of the resource, with an optional reference to an organization that is the authority for the version or release.

Content Model

```
<!ELEMENT Version (Organization?,Date?) >
```

Attributes

<!ATTLIST Version		
title	CDATA	#IMPLIED
description	CDATA	#IMPLIED
full-version	CDATA	#IMPLIED
major-version	CDATA	#IMPLIED
minor-version	CDATA	#IMPLIED
url	%url;	#IMPLIED>

Date Element

Date and time stamp in ISO 8601 format. The role of the date and time stamp may also be specified.

Content Model

```
<!ELEMENT Date EMPTY >
```

Attributes

<!ATTLIST Date		
datetime	%datetime;	#REQUIRED
role (created valid available issued modified other)		#IMPLIED
other	CDATA	#IMPLIED>

Content Element

A general element used to supply any type of parsed or unparsed content.

Content Model

```
<!ELEMENT Content (#PCDATA)>
```

Attributes

<!ATTLIST Content		
id	ID	#IMPLIED
name	CDATA	#IMPLIED
description	CDATA	#IMPLIED
content-type	CDATA	#IMPLIED
role	CDATA	#IMPLIED>

Identifier Element

Standardized identifier of a resource (e.g., ISBN).

Content Model

```
<!ELEMENT Identifier    (#PCDATA)  >
```

Attributes

```
<!ATTLIST Identifier
  context      CDATA      #IMPLIED
  refid        CDATA      #IMPLIED
  url           %url;      #IMPLIED>
```

Notes

<u>Attributes</u>	<u>Explanation</u>
context	e.g., ISBN
refid	

History Element

A list of history events.

Content Model

```
<!ELEMENT History (History-event*) >
```

Attributes

```
<!ATTLIST History
  id          ID          #IMPLIED
  title       CDATA       #IMPLIED
  description  CDATA       #IMPLIED
  type        CDATA       #IMPLIED>
```

History-event Element

An event in a history.

Content Model

```
<!ELEMENT History-event (Date?, Contributor*,
  History-event-description?, %links;) >
```

Attributes

```
<!ATTLIST History-event
  id          ID          #IMPLIED
  title       CDATA       #IMPLIED
  description  CDATA       #IMPLIED
  type        CDATA       #IMPLIED>
```

History-event-description Element

The description of an event in a history.

Content Model

```
<!ELEMENT History-event-description (#PCDATA) >
```

Authority Element

Identification by URL of a namespace, controlled vocabulary source, or similar resource.

Content Model

```
<!ELEMENT Authority (Authority-url*,Authority-domain*) >
```

Attributes

```
<!ATTLIST Authority
  id          ID          #IMPLIED
  title       CDATA      #IMPLIED
  description  CDATA      #IMPLIED
  type        CDATA      #IMPLIED
>
```

Notes

Attributes

site-url	optional link to authority site
list-url	optional link to get text list of vocabulary
access-url	optional link to access item info by appending attribute text to url

Explanation

Authority-url Element

A link to a resource that supports the authority.

Content Model

```
<!ELEMENT Authority-url EMPTY >
```

Attributes

```
<!ATTLIST Authority-url
  id          ID          #IMPLIED
  title       CDATA      #IMPLIED
  description  CDATA      #IMPLIED
  type (website | vocabulary-list | identifier-access | other) #IMPLIED
  other-type   CDATA      #IMPLIED
  content-type  %contenttype; #IMPLIED
  url          %url;      #REQUIRED>
```

Authority-domain Element

An element and attribute whose value is controlled by the authority.

Content Model

```
<!ELEMENT Authority-domain EMPTY >
```

Attributes

```
<!ATTLIST Authority-domain
  title          CDATA          #IMPLIED
  description     CDATA          #IMPLIED
  element-name    CDATA          #REQUIRED
  attribute-name  CDATA          #REQUIRED
  class-names     CDATA          #IMPLIED>
```

Notes

Attributes

element-name
attribute-name
class-names

Explanation

name of element to apply authority
name of attribute to apply authority
optional list of class attributes
limiting applicability of authority

Linking Elements

Attribute Element

See **Basic Attributes and Elements** for the content models of **Attribute** and **Attribute-list** elements.

Cross-reference Element

Used to link to external resources identified by database and identifier.

Content Model

```
<!ELEMENT Cross-reference EMPTY>
```

Attributes

```
<!ATTLIST Cross-reference
  id            ID            #IMPLIED
  context       CDATA         #IMPLIED
  database      CDATA         #IMPLIED
  identifier     CDATA         #IMPLIED
  identifier-type CDATA        #IMPLIED
  title         CDATA         #IMPLIED
  behavior      CDATA         #IMPLIED
  href          CDATA         #IMPLIED
  role          CDATA         #IMPLIED>
```


Notes*

<u>Attributes</u>	<u>Explanation</u>
id	
context	namespace for databases/ontologies
database	abbreviation for database
identifier	resource identifier
identifier-type	e.g., accession or GUID
title	displayable title
behavior	specify action
href	specify access
role	controlled vocabulary fo xref types

- **Cross-reference** may be used as a linking element in place of the previous qualifier dbxref entry.

Locator Element

Link resource used to identify destination of link.

Content Model

<!ELEMENT Locator EMPTY>

Attributes

```
<!--ATTLIST Locator
  id          ID          #IMPLIED
  xml-link    CDATA      #FIXED "LOCATOR"
  role        CDATA      #IMPLIED
  href        CDATA      #REQUIRED
  title       CDATA      #IMPLIED
  rel         CDATA      #IMPLIED
  rev         CDATA      #IMPLIED
  show        (embed|replace|new) "embed"
  actuate     (auto|user) "user"
  behavior    CDATA      #IMPLIED-->
```

Link Element

Simple link used to identify a single destination.

Content Model

<!ELEMENT Link EMPTY>

Attributes

```
<!--ATTLIST Link
  id          ID          #IMPLIED
  xml-link    CDATA      #FIXED "SIMPLE"
  role        CDATA      #IMPLIED
  href        CDATA      #REQUIRED
  title       CDATA      #IMPLIED
  rel         CDATA      #IMPLIED
  rev         CDATA      #IMPLIED-->
```

show	(embed replace new) "embed"
actuate	(auto user) "user"
behavior	CDATA #IMPLIED>

Notes

<u>Attribute</u>	<u>Explanation</u>
id	unique identifier for accessing and actuating link
role	specifies the function of the link
href	specifies URL to be accessed when link is actuated; may reference another element in same document
title	displayable caption for the link
rel	specifies the relationship of the linking resource to its destination
rev	specifies relationship of the destination resource to this resource
show	specifies display and processing behavior applied to destination resource
Embed	embed the target resource in the body of the resource
Replace	substitute the target for the current resource
New	use a new context to display the target resource
Actuate	indicates whether link should be fired when document is loaded (auto) or only when selected (user)
Behavior	specific instructions for actuating the link

Extended-link Element

A one-to-many linking structure that points to elements or cross-references.

Content Model

```
<!ELEMENT Extended-link (Locator|Cross-reference)*>
```

Attributes

```
<!ATTLIST Extended-link
  id ID #IMPLIED
  xml-link CDATA #FIXED "EXTENDED"
  inline CDATA #FIXED "TRUE"
  content-title CDATA #IMPLIED
  content-role CDATA #IMPLIED
  role CDATA #IMPLIED
  href CDATA #IMPLIED
  title CDATA #IMPLIED
  rel CDATA #IMPLIED
  rev CDATA #IMPLIED
  show (embed|replace|new) "embed"
  actuate (auto|user) "user"
  behavior CDATA #IMPLIED>
```

Document-link Element

A link element that points to an external document.

Content Model

<!ELEMENT Document-link EMPTY>

Attributes

```
<!ATTLIST Document-link
  id          ID          #IMPLIED
  xml-link    CDATA       #FIXED "DOCUMENT"
  role        CDATA       #IMPLIED
  href        CDATA       #REQUIRED
  title       CDATA       #IMPLIED
  rel         CDATA       #IMPLIED
  rev         CDATA       #IMPLIED
  mime-type   CDATA       #IMPLIED
  show        (embed|replace|new) "embed"
  actuate     (auto|user) "user"
  behavior    CDATA       #IMPLIED>
```

Group-link Element

A one-to-many linking structure that points to a set of documents.

Content Model

<!ELEMENT Group-link (Document-link)*>

Attributes

```
<!ATTLIST Group-link
  id          ID          #IMPLIED
  xml-link    CDATA       #FIXED "GROUP"
  steps       CDATA       #IMPLIED
  role        CDATA       #IMPLIED
  href        CDATA       #IMPLIED
  title       CDATA       #IMPLIED
  rel         CDATA       #IMPLIED
  rev         CDATA       #IMPLIED
  show        (embed|replace|new) "embed"
  actuate     (auto|user) "user"
  behavior    CDATA       #IMPLIED>
```

Links Element

Any number of linking elements.

Content Model

<!ELEMENT Links %links;>

3.3 BSML Element

A complete BSML document begins with the standard XML processing instruction and encloses all of its content between **<Bsml>** tags. Within the **<Bsml>** tags, the document is divided into three discrete sections:

- **Definitions** – encoding of data
- **Research** – encoding of research descriptions [3]
- **Display** – encoding of display widgets for visualization of data

Content Model

```
<!ELEMENT Bsml (Attribute*, Info*, Resource* [3],  
                Definitions?,  
                Research?[3],  
                Display?) >
```

Notes:

The **Definitions** section may be used without a **Display** section if the goal is simply to encode sequence data; for convenience, a few elements in the **Definitions** section have display-related attributes, but these attributes are never required. Thus a BSML document may be used to convey data independently of a BSML browser, and the document may be read and processed by any suitable XML-parsing software.

The **Display** section provides information that is mainly relevant for the graphical display of bioinformatic data and for accessing links in other network resources. Much of the content of the **Display** section only has meaning for a browser that works specifically with BSML and has been customized to display the widgets defined in the BSML standard.

General information (document author, creation date, etc.) may be attached to the entire document by using **Attribute** and **Info** elements as content of the **Bsml** element.

3.4 Definitions

Data are contained in the **Definitions** section of a BSML document. There are six main **Definitions** subsections: **Genomes**, **Sequences**, **Isoforms**, **Tables**, **Sets**, and **Networks**.

Content Model

```
<!ELEMENT Definitions (Attribute*,
                        Genomes?[3],
                        Sequences?,
                        Isoforms?[3],
                        Sets?,
                        Tables?,
                        Networks?[3])>
```

Notes

Most elements in this section may have additional information attached to them by adding **Attribute**, **Resource**, or **Link** elements.

Genomes Element [3]

The **Genomes** element defines a collection of genomes, each defined by a **Genome** element.

Content Model

```
<!ELEMENT Genomes (Attribute*, Genome*, Resource*, %links;)> [3]
```

Genome Element [3]

This element refers to the organism and defines chromosomes and resources.

Content Model

```
<!ELEMENT Genome (Attribute*, (Organism | Organism-clone | Cell-line)?,
                        Chromosome*, Extrachromosomal-sequence*, Resource*, %links;) >[3]
```

Attributes

```
<!ATTLIST Genome
%attr;
autosomal-chromosome-count    %integer; #IMPLIED
sex-chromosome-count          %integer; #IMPLIED
ploidy-count                   %integer; #IMPLIED
distinct-chromosome-count     %integer; #IMPLIED
total-chromosome-count        %integer; #IMPLIED >
```

Notes

<u>Attribute</u>	<u>Explanation</u>
autosomal-chromosome-count	Number of autosomal chromosomes

sex-chromosome-count	Number of distinct sex chromosome
ploidy-count	Multiple for autosomal chromosomes
distinct-chromosome-count	Autosomal + sex
total-chromosome-count	Ploidy * autosomal + sex

Organism Element

Note: See the **Research** element and its children for content models.

Organism-clone Element

Note: See the **Research** element and its children for content models.

Cell-line Element

Note: See the **Research** element and its children for content models.

Chromosome Element [3]

This element identifies a chromosome and any number of cytobands. Chromosomal data may be accessed by including the **Seq-data-import** element.

Content Model

```
<!ELEMENT Chromosome (Cytoband*, Seq-data-import?)> [3]
```

Attributes

```
<!ATTLIST Chromosome
  id             ID             #IMPLIED
  name           CDATA          #IMPLIED
  number         CDATA          #IMPLIED
  length         %integer;      #IMPLIED
  circular        %yesorno;     #IMPLIED
  autosomal      %yesorno;     #IMPLIED>
```

Notes

<u>Attribute</u>	<u>Explanation</u>
id	id
name	chromosome name
number	chromosome number
length	chromosome length
circular	circular (1) or linear (0)
autosomal	autosomal (1) or sex (0)

Extrachromosomal-sequence Element [3]

This element describes extrachromosomal sequence(s) and any number of associated cytobands. Sequence data may be accessed by including the **Seq-data-import** element.

Content Model

<!ELEMENT Extrachromosomal-sequence (Cytoband*, Seq-data-import?)> [3]

Attributes

```
<!ATTLIST Extrachromosomal-sequence
  id             IDREF          #IMPLIED
  type           CDATA          #IMPLIED
  name           CDATA          #IMPLIED
  number         CDATA          #IMPLIED
  length         %integer;      #IMPLIED
  circular       %yesorno;      #IMPLIED>
```

Notes

<u>Attribute</u>	<u>Explanation</u>
id	id
type	type of extrachromosomal sequence
name	name of extrachromosomal sequence
number	extrachromosomal sequence number
length	length of extrachromosomal sequence
circular	circular or linear

Cytoband Element [3]

This element describes a cytoband.

Content Model

<!ELEMENT Cytoband EMPTY>

Attributes

```
<!ATTLIST Cytoband
  id             ID              #IMPLIED
  band-name      CDATA           #IMPLIED
  major-band     CDATA           #IMPLIED
  minor-band     CDATA           #IMPLIED
  band-color     CDATA           #IMPLIED
  band-type      (euchromatin|telomere|centromere|heterochromatin) #IMPLIED
  band-start-pos %integer;       #IMPLIED
  band-end-pos   %integer;       #IMPLIED>
```

Notes

<u>Attribute</u>	<u>Explanation</u>
band-name	Complete name, e.g., P31.1
major-band	Major band name, with arm, e.g., P31
minor-band	Minor band name, e.g., 1
band-color	Staining indicator
band-type	Uses controlled vocabulary above

Sequences Element

The **Sequences** element contains references to any number of DNA, RNA, or protein sequences. There are two elements that are used to define sequences: **Sequence** and **Sequence-import**. A **Sequence** element defines a sequence in the current BSML document by setting various attributes. A **Sequence-import** element refers via a URL (Uniform Resource Location) to a document containing the sequence information. **Segment-set** is used to refer to fragments or complete extents of internally or externally defined sequences.

Content Model

```
<!ELEMENT Sequences (Attribute*,(Sequence|Sequence-import)*,
                      Segment-set*[3], Resource*, %links;)>
```

Segment-set Element [3]

The **Segment-set** element defines the relationship among a group of two or more sequence fragments represented by **Segment** elements.

Content Model

```
<!ELEMENT Segment-set (Attribute*, Segment+, Resource*, %links;)> [3]
```

Attributes

```
<!ATTLIST Segment-set
    %attrs;
    seg-set-type    CDATA          #IMPLIED >
```

*Notes:

<u>Attribute</u>	<u>Explanation</u>
seg-set-type	type of set (alignment, etc.)

Segment Element

The **Segment** element describes a region of a sequence.

Content Model

```
<!ELEMENT Segment (Attribute*,Resource*,%links;) >
```

Attributes

```
<!ATTLIST Segment
    %attrs;
    seg-source-type    CDATA          #IMPLIED
    seg-source         CDATA          #IMPLIED
    seg-id             CDATA          #IMPLIED
    seg-url            CDATA          #IMPLIED
    seg-role           CDATA          #IMPLIED
    seg-start          %integer;      #IMPLIED
    seg-end            %integer;      #IMPLIED
    seg-on-complement  %yesorno;      #IMPLIED
```


seg-translated %yesorno; #IMPLIED>

*Notes:

<u>Attribute</u>	<u>Explanation</u>
seg-source-type	type may be external database
seg-source	source name
seg-id	identifier
seg-url	url, if appropriate
seg-role	role played by sequence fragment
seg-start	start of fragment
seg-end	end of fragment
seg-on-complement	on complement if true
seg-translated	translated if true

Sequence Element

Content Model

```
<!ELEMENT Sequence (Attribute*,Feature-tables?,
                    (Seq-data|Seq-data-import)?,Numbering?,
                    Modification* [3], Segment*, Resource*, %links;)>
```

Attributes

```
<!ATTLIST Sequence
  %attrs;
  locus          CDATA          #IMPLIED
  ic-acckey       CDATA          #IMPLIED
  local-acckey    CDATA          #IMPLIED
  db-source       %dbsource;     #IMPLIED
  length          CDATA          #IMPLIED
  end5hang5       %integer;      #IMPLIED
  end3hang5       %integer;      #IMPLIED
  end5phos        %yesorno;      #IMPLIED
  end3phos        %yesorno;      #IMPLIED
  genomeref      IDREF          #IMPLIED
  trans-table     CDNA           #IMPLIED
  segmenttype     %segment-opts;  "sequence"
  dnatype         (genomic | cdna) #IMPLIED
  representation  %repr-opts;    "raw"
    (not-set|virtual|raw|segmented|constructed|
    reference|consensus|map|derived|other)
  molecule (mol-not-set|dna|rna|aa|na|other-mol) "dna"
  topology (top-not-set|linear|circular|tandem|top-other) "linear"
  strand (std-not-set|ss|ds|mixed|std-other) #IMPLIED
  refs          IDREFS #IMPLIED
  alignment-sequence IDREFS #IMPLIED
  alignment-position CDATA #IMPLIED
  alignment-inverted CDATA #IMPLIED>
```

*Notes:

<u>Attribute</u>	<u>Explanation</u>
locus	sequence name
ic-acckey	internatl. collaboration

local-acckey	local/private accession key
db-source	database source of sequence
length	integer value total length
end5hang5	5' overhang on 5' end
end3hang5	5' overhang on 3' end
end5phos	yes= 5' end phosphorylated
end3phos	yes= 3' end phosphorylated
genomeref	reference to the Genome of the organism
trans-table	title of the NCBI translation table
segmenttype	type of sequence being represented
dnatype	genomic or cdna
representation	defines sequence representation types (as described by NCBI)
molecule	type of molecule (dna, etc.)
topology	molecule shape (linear, circular, etc.)
strand	strand=ds for dna and ss otherwise, if not specified
refs	element cross-references - including references, features, other sequences, etc. - if sequence is segmented, constructed, etc., refs includes ids of sequences or segments comprising the complete sequence; the sequences referred to may indicate their status by using the alignment-sequences and alignment- positions attributes
alignment-sequence	aligned to sequences
alignment-position	positions on aligned
alignment-inverted	0=ascending, 1=descending

Sequence-import Element

This element identifies a **Sequence** element to be imported from another BSML document.

Content Model

```
<!ELEMENT Sequence-import (Attribute*, Resource*, %links;)>
```

Attributes

```
<!ATTLIST Sequence-import
  type      CDATA      #FIXED  "bsml"
  source     %url;      #REQUIRED
  id         ID         #IMPLIED>
```

*Notes:

<u>Attribute</u>	<u>Explanation</u>
type	only supports BSML document fragments import
source	file name or URL
id	id identifier selects one sequence from a set; if bsml import, identifier = id

if other, identifier = primary accession
key

Seq-data-import Element

Specification of a file or url from which sequence data are to be obtained.

Content Model

```
<!ELEMENT Seq-data-import (Attribute*)>
```

Attributes

```
<!ATTLIST Seq-data-import>
%attrs;
format      CDATA      #IMPLIED
source      %url;      #IMPLIED
identifier   CDATA      #IMPLIED
start-pos   %integer;  #IMPLIED
length      %integer;  #IMPLIED
encrypted    %yesorno;  #IMPLIED>
```

*Notes

<u>Attribute</u>	<u>Explanation</u>
format	ascii, genbank, embl, fasta, bsml or one of the above for pure data
source	if none, librarian needed
identifier	identifier selects one sequence from a set; if bsml import, identifier = id if other, identifier = primary accession key
encrypted	if yes, decryption key needed

In addition to an ASCII file, the format may be one of the following:

```
IUPACna : IUPAC 1 letter codes, no spaces
IUPACaa : IUPAC 1 letter codes, no spaces
NCBI2na : 00=A, 01=C, 10=G, 11=T (4 bases per byte)
NCBI4na : 1 bit each for agct (2 bases per byte)
          0001=A, 0010=C, 0100=G, 1000=T/U
          0101=Purine, 1010=Pyrimidine, etc
```

Seq-data Element

Sequence data specified using IUPAC na and aa single-letter codes. All other characters and whitespace are ignored.

Content Model

```
<!ELEMENT Seq-data (#PCDATA)>
```

*Notes:

Seq-data	the sequence data dna/rna=IUPACna, protein=IUPACaa
----------	---

Modification Element [3]

Specification of a modification to a base or residue.

Content Model

```
<!ELEMENT Modification (Attribute*,Resource*,%links;) > [3]
```

Attributes

```
<!ATTLIST Modification
  %attrs;
  position %integer; #REQUIRED
  change   CDATA     #REQUIRED
  source    CDATA     #IMPLIED>
```

Notes*

<u>Attributes</u>	<u>Explanation</u>
position	Location on the sequence
change	Nature of modification
source	If controlled vocabulary, authority, e.g., WIPO ST.25 Modification type

Location Types

The location entity provides reference to either type of location.

```
<!ENTITY % location "(Site-loc|Interval-loc)">
```

Site-loc Element

Represents a single position on a sequence.

Content Model

```
<!ELEMENT Site-loc (Attribute*,Resource*,%links;) >
```

Attributes

```
<!ATTLIST Site-loc
  %attrs;
  sitepos      CDATA     #REQUIRED
  fuzzypos     CDATA     #IMPLIED [3]
  complement    %yesorno; "0"
  strand-unknown %yesorno; "0">
```

*Notes:

<u>Attribute</u>	<u>Explanation</u>
sitepos	integer position on sequence
fuzzypos	site position plus or minus this value
complement	on complement if yes
strand-unknown	strand information not known if yes

Interval-loc Element

Represents an interval or range on the sequence, indicating either the range of the feature or the region in which a point is contained.

Content Model

```
<!ELEMENT Interval-loc (Attribute*,Resource*,%links;) >
```

Attributes

```
<!ATTLIST Interval-loc
  %attrs;
  startpos          CDATA          #REQUIRED
  fuzzystart        CDATA          #IMPLIED [ 3 ]
  endpos            CDATA          #REQUIRED
  fuzzyend          CDATA          #IMPLIED
  startopen         %yesorno;      "0"
  endopen           %yesorno;      "0"
  onepos            %yesorno;      "0"
  complement        %yesorno;      "0"
  strand-unknown    %yesorno;      "0"
  segment-ref       IDREF          #IMPLIED [ 3 ]>
```

*Notes:

Attributes

startpos
fuzzystart
endpos
fuzzyend
startopen
endopen
onepos
complement
strand-unknown
segment-ref

Explanation

integer start of range
start plus or minus this value
integer end of range
end plus or minus this value
open if yes
open if yes
one position if yes
on complement if yes
true if yes
pointer to **Segment-set** element or **Segment**

Qualifier Element

A **Qualifier** uses value-type and value attributes to qualify a **Feature**. The value-type may represent a controlled vocabulary if an authority is specified as a **Resource** for the **Feature** or **Feature-table**.

Content Model

```
<!ELEMENT Qualifier (Attribute*,Resource*,%links;) >
```

Attributes

```
<!ATTLIST Qualifier
  %attrs;>
```

Feature Element

An annotation on a sequence is represented by a **Feature** element, which may contain any number of **Qualifier** and **location** elements. If **join** is true, the first **location** element defines the complete range spanned by the join and the additional **location** elements define each segment of the join.

Content Model

```
<!ELEMENT Feature (Attribute*, (%location;|Qualifier)*,
                      Resource*[3], %links;)>
```

Attributes

```
<!ATTLIST Feature
  %attrs;
  display-auto    %yesorno;    #IMPLIED
  auto-view       IDREF        #IMPLIED
  refs            IDREFS       #IMPLIED
  join            %yesorno;    #IMPLIED [3]
  group-type      CDATA        #IMPLIED [3]
  segment-ref     IDREF        #IMPLIED [3]>
```

*Notes:

<u>Attributes</u>	<u>Explanation</u>
display-auto	if yes, create and show default display widget
auto-view	view to which to attach display widget, if not defined, feature added to first view for this seq;
refs	feature type description encoded as value-type core attribute
join	element cross-references, including references, other features, seqs, etc.
group-type	if true, this feature is a join
segment-ref	if the feature is a member of a group use this attribute to identify the group pointer to Segment-set or Segment

Reference Element

Content Model

```
<!ELEMENT Reference
  (Attribute*,RefAuthors?,RefTitle?,RefJournal?,Resource*,%Links;)>
```

Attributes

```
<!ATTLIST Reference
  %attrs;
  dbxref          CDATA        #IMPLIED
  refs            IDREFS       #IMPLIED>
```

*Notes

<u>Attribute</u>	<u>Explanation</u>
dbxref	reference to external source, e.g., medline
refs	element cross-references, including references, other features, seqs, etc.

- References may also be linked to external resources such as medline references by using the ID of a link in refs

RefAuthors Element

Authors of a bibliographic entry.

Content Model

```
<!ELEMENT RefAuthors (#PCDATA) >
```

RefTitle Element

Title of a bibliographic entry.

Content Model

```
<!ELEMENT RefTitle (#PCDATA) >
```

RefJournal Element

Journal of a bibliographic entry.

Content Model

```
<!ELEMENT RefJournal (#PCDATA) >
```

Digest Set Element

Content Model

```
<!ELEMENT Digest-set (Attribute*, Restriction-enzyme*,
                      Resource*[3], %links;)>
```

Attributes

```
<!ATTLIST Digest-set
  %attrs;
  fragments CDATA #IMPLIED
  units CDATA #IMPLIED
  start-pos %integer; #IMPLIED
  end-pos %integer; #IMPLIED
  min-cuts %integer; #IMPLIED
  max-cuts %integer; #IMPLIED>
```

*Notes

<u>Attribute</u>	<u>Explanation</u>
------------------	--------------------

fragments	comma-separated list
units	if fragments
- start and end position used	if only part of sequence searched -
start-pos	else seq start
end-pos	else seq end
- for documentation, if generated under search conditions -	
min-cuts	allowed in set
max-cuts	allowed in set

Restriction-enzyme Element

Contains sites for a particular restriction enzyme, with **title** = enzyme name.

Content Model

```
<!ELEMENT Restriction-enzyme EMPTY>
```

Attributes

```
<!ATTLIST Restriction-enzyme>
%attrs;
source          CDATA          #IMPLIED
recognition-seq CDATA          #IMPLIED
offset          %integer;      "0"
overhang        %integer;      "0"
sites           CDATA          #IMPLIED>
```

*Notes

<u>Attribute</u>	<u>Explanation</u>
source	sequence or organism
recognition-seq	base pattern
offset	cut position relative to the 5' end of pattern
overhang	5' overhang
sites	comma-separated list
	site positions are indicated by the base 3' of the cut; negative numbers indicate recognition on minus strand for non-palindromic enzymes

* fragments may be used in place of restriction-enzymes to indicate directly the sizes of the fragments, which are otherwise computed from the restriction sites

Feature-tables Element

Each **Sequence** may have any number of feature tables.

Content Model

```
<!ELEMENT Feature-tables (Attribute|Feature-table|Feature-group[3])*>
```

Attributes

```
<!ATTLIST Feature-tables
```



```
%attrs;>
```

Feature-table Element

Each **Feature-table** is usually given a distinct **class** identifier to indicate the type of features or other elements contained in the set.

Content Model

```
<!ELEMENT Feature-table ((Attribute*,(Reference|Feature|Digest-set)*,Resource*, %links;)>
```

Attributes

```
<!ATTLIST Feature-table
  %attrs; >
```

Feature-group Element [3]

Defines a set of related features such as the exons comprising a gene.

Content Model

```
<!ELEMENT Feature-group (Attribute*,Feature-group-member*,Resource*,%links;)> [3]
```

Attributes

```
<!ATTLIST Feature-group
  %attrs;
  group-set CDATA #IMPLIED>
```

Notes

Attributes

group-set

Explanation

link together a set of related groups

Feature-group-member Element [3]

Defines one of the members of a **Feature-group**.

Content Model

```
<!ELEMENT Feature-group-member (#PCDATA) > [3]
```

Attributes

```
<!ATTLIST Feature-group-member
  %attrs;
  featref IDREF #REQUIRED
  feature-type CDATA #IMPLIED
  group-type CDATA #IMPLIED>
```

Notes

Attributes

featref
feature-type

group-type

Explanation

Refers to a **Feature** in a **Feature-table**
Indicates the type of the feature,
e.g., exon
Indicates the group type

Numbering Element

Internally, the length of every sequence is expressed as a positive integer. Sequences may specify their unit of measurement (bp, cM, etc.) through the use of a **Numbering** element, which also may define a relative numbering basis or a linear transformation rule for converting internal sequence positions to display values.

Content Model

```
<!ELEMENT Numbering EMPTY >
```

Attributes

```
<!ATTLIST Numbering
  %attrs;
  seqref          IDREF          #IMPLIED [3]
  use-numbering   %yesorno;      #IMPLIED
  type (continuous|real|enumerated|reference) #IMPLIED
  units          CDATA          #IMPLIED
  a              %real;         #IMPLIED
  b              %real;         #IMPLIED
  dec-places     %integer;      #IMPLIED
  refnum         %integer;      #IMPLIED
  has-zero       %yesorno;      #IMPLIED
  ascending      %yesorno;      #IMPLIED
  names          CDATA          #IMPLIED
  from-aligns    %yesorno;      #IMPLIED
  aligns         IDREF          #IMPLIED>
```

Isoforms Element [3]

Use **Isoforms** for alleles, mutations, SNPs, case records, and pedigrees.

Content Model

```
<!ELEMENT Isoforms (Isoform-set | Case-set | Phenotype-set |
                    Genotype-set | Pedigree-set)* > [3]
```

Isoform-set Element [3]

This element contains a set of **Isoform** elements.

Content Model

```
<!ELEMENT Isoform-set (Attribute*,
                      (Organism | Organism-clone | Cell-line)?,
```

Isoform*, Resource*, %links;)> [3]

Attributes

```
<!ATTLIST Isoform-set
  %attrs;>
```

Isoform Element [3]

An **Isoform** element indicates a change (insertion, deletion, transversion, etc.) in a sequence.

Content Model

```
<!ELEMENT Isoform (Attribute*, Resource*, %links;)> [3]
```

Attributes

```
<!ATTLIST Isoform
  %attrs;
  seqref      IDREF      #IMPLIED
  phenoref    IDREF      #IMPLIED
  url         %url;      #IMPLIED
  location    CDATA      #IMPLIED
  change      CDATA      #IMPLIED>
```

Phenotype-set element [3]

A set of **phenotype** descriptions with respect to a particular organism.

Content Model

```
<!ELEMENT Phenotype-set (Attribute*,
  (Organism | Organism-clone | Cell-line)?,
  Phenotype*, Resource*, %links;) > [3]
```

Attributes

```
<!ATTLIST Phenotype-set
  %attrs;
  population    CDATA #IMPLIED
  source        CDATA #IMPLIED
  frequency-total CDATA #IMPLIED>
```

Phenotype Element [3]

Description of a phenotype using the **value-type** and **value** attributes.

Content Model

```
<!ELEMENT Phenotype (Attribute*, Resource*, %links;) > [3]
```

Attributes

```
<!ATTLIST Phenotype
```

```
%attrs;
frequency CDATA #IMPLIED>
```

Case-set Element [3]

A set of **Case** elements.

Content Model

```
<!ELEMENT Case-set (Attribute*,
                    (Organism | Organism-clone | Cell-line)?,
                    Case*, Resource*, %links;)> [3]
```

Attributes

```
<!ATTLIST Case-set
  %attrs;>
```

Case Element [3]

An individual case or individual characterized by geotypical and phenotypical descriptions.

Content Model

```
<!ELEMENT Case (Attribute*,(Organism|Organism-clone|Cell-line)?,
                Isoform*, Life-event*, Genotype-set*,
                Resource*, %links;)> [3]
```

Attributes

```
<!ATTLIST Case
  %attrs;
  sex          (male|female|other|unknown)      #IMPLIED
  propositus   %yesorno;                        #IMPLIED>
```

Genotype-set element [3]

A set of **Genotype** descriptions with respect to a particular organism.

Content Model

```
<!ELEMENT Genotype-set (Attribute*,
                        (Organism | Organism-clone | Cell-line)?,
                        Genotype*, Resource*, %links;) > [3]
```

Attributes

```
<!ATTLIST Genotype-set
  %attrs;
  population      CDATA #IMPLIED
  source          CDATA #IMPLIED
  frequency-total CDATA #IMPLIED>
```

Genotype Element [3]

A description of a genotype.

Content Model

```
<!ELEMENT Genotype (Haplotype*)> [3]
```

Attributes

```
<!ATTLIST Genotype
  %attrs;
  homozygous      %yesorno;    #IMPLIED
  recessive       %yesorno;    #IMPLIED
  sex-linked      %yesorno;    #IMPLIED
  phenoref        IDREF        #IMPLIED
  affected        %yesorno;    #IMPLIED
  population      CDATA        #IMPLIED
  source          CDATA        #IMPLIED
  frequency-total CDATA        #IMPLIED>
```

Haplotype Element [3]

A description of a haplotype.

Content Model

```
<!ELEMENT Haplotype EMPTY> [3]
```

Attributes

```
<!ATTLIST Haplotype
  %attrs;
  frequency CDATA #IMPLIED>
```

Life-event Element [3]

An event in the life of an individual.

Content Model

```
<!ELEMENT Life-event EMPTY > [3]
```

Attributes

```
<!ATTLIST Life-event
  %attrs;
  life-event-type
    (birth|marriage|death|abortion-or-stillbirth|other-life-event)
    #REQUIRED
  year      %integer; #REQUIRED
  month     %integer; #IMPLIED
  day       %integer; #IMPLIED>
```

Children Element [3]

The children produced by a mating as part of describing a pedigree.

Content Model

```
<!ELEMENT Children (Case*) > [3]
```

Mating Element [3]

A mating between two individuals in a pedigree.

Content Model

```
<!ELEMENT Mating (Life-event?, Children?) > [3]
```

Attributes

```
<!ATTLIST Mating
  %attrs;
  maleref          IDREF          #REQUIRED
  femaleref        IDREF          #REQUIRED
  consanguinal     %yesorno;      #IMPLIED>
```

Pedigree Element [3]

A pedigree describing a pattern of genotypes and phenotypes in a set of related individuals.

Content Model

```
<!ELEMENT Pedigree (Attribute*, (Organism|Organism-clone|Cell-line)?,
Mating*, Resource*, %links;) [3]
```

Attributes

```
<!ATTLIST Pedigree
  %attrs;>
```

Pedigree-set Element [3]

A set of pedigrees.

Content Model

```
<!ELEMENT Pedigree-set (Pedigree*)> [3]
```

Attributes

```
<!ATTLIST Pedigree-set
  %attrs;>
```

Tables Element

A set of tabular and hierarchical data representations for various types of data.

Content Model

```
<!ELEMENT Tables (Attribute|Table|Motif-table
                  | Sequence-search-table [3]
                  | Multiple-alignment-table [3]
                  | Seq-pair-alignment
                  | Alignment-point-set
                  | PCR-primer-table [3]
                  | Table-import)* >
```

Attributes

```
<!ATTLIST Tables
  %attrs; >
```

HTML Table Element

The BSML definition for a **Table** follows the HTML 4.0 specifications.

Table-import Element

A data table defined with reference to an external data **source** or by defining data in a set of **Table-data** elements.

Content Model

```
<!ELEMENT Table-import (Attribute*,Table-data*, Resource*[3], %links;)>
```

Attributes

```
<!ATTLIST Table-import
  %attrs;
  format          CDATA          #IMPLIED
  read-field-desc %yesorno;      #IMPLIED
  column-numbers  CDATA          #IMPLIED
  column-headers  CDATA          #IMPLIED
  column-starts   CDATA          #IMPLIED
  column-widths   CDATA          #IMPLIED
  header-count    %integer;      #IMPLIED
  display-widths  CDATA          #IMPLIED
  alignments      CDATA          #IMPLIED
  font            IDREF          #IMPLIED
  add-row-numbers %yesorno;      #IMPLIED
  locked-columns  %integer;      #IMPLIED
  source          %url;          #REQUIRED
  refs            IDREFS         #IMPLIED>
```

*Notes

<u>Attribute</u>	<u>Explanation</u>
format	acceptable formats=bsml html txt sdf db

	if data in Table-data elements, no format needed
read-field-desc	use for txt,sdf if read-field-desc=yes, read field descriptions at start of file = Num (number of fields on first line) FieldName,Type(=T,N,L),Width,Decimals with one line per field list of columns to include; use all if not defined
column-numbers	
-- comma-separated list of column headings --	
column-headers	use for txt,sdf
-- comma-separated lists of start positions and field widths--	
column-starts	used for txt
column-widths	used for txt
header-count	text skip lines
display-widths	comma-separated list of column widths for display in grid viewer
alignments	comma-separated list of L (left=text) or R (right=numeric) for each column
font	font for grid
add-row-numbers	number grid rows
locked-columns	# to lock at left
source	file/remote
refs	element cross-references

Table-data Element

Locally supplied data that may be included with a **Table**-import element.

Content Model

```
<!ELEMENT Table-data EMPTY >
```

Attributes

```
<!ATTLIST Table-data
  %attrs;>
```

*Notes

- * use title to name table row, if appropriate
- * uses **value** for comma-separated data table row list

Motif Table Related Elements

Motif-data Element

Describes one set of data for a **Motif-table**.

Content Model

```
<!ELEMENT Motif-data (Motif-data*)>
```

Attributes


```

<!ATTLIST Motif-data
  %attrs;
  one-element %yesorno; "0"
  positions   CDATA      #REQUIRED
  widths      CDATA      #IMPLIED
  strands     CDATA      #IMPLIED
  titles       CDATA      #IMPLIED
  values       CDATA      #IMPLIED
  markers      CDATA      #IMPLIED
  overhangs    CDATA      #IMPLIED
  refs         CDATA      #IMPLIED>

```

*Notes

<u>Attribute</u>	<u>Explanation</u>
one-element	by default, multi-element is true
positions	comma-separated list of positions
widths	comma-separated list of motif widths
strands	comma-separated list of plus=1,minus=0
titles	comma-separated list of titles
values	comma-separated list of values
markers	comma-separated 5 prime offsets
overhangs	comma-separated overhang on minus std
refs	space-separated list of IDs
* if hierachical, top level owns next level, etc.	
* if flat, use comma-separated lists	
* if flat and site series for 1-element motif, one-element=true	

Sequence-motif Element

Describes one sequence participating in a **Motif-table**.

Content Model

```
<!ELEMENT Sequence-motif (Interval-loc*,Motif-data*)>
```

Attributes

```

<!ATTLIST Sequence-motif
  %attrs;
  display-auto %yesorno; #IMPLIED
  auto-view    IDREF      #IMPLIED
  seqref       IDREF      #REQUIRED
  alignment    %integer;   #IMPLIED
  startpos     %integer;   #IMPLIED
  endpos       %integer;   #IMPLIED
  refs         CDATA      #IMPLIED>

```

*Notes

<u>Attribute</u>	<u>Explanation</u>
display-auto	if display-auto, create default display widget and show
auto-view	view to which display widget is attached,if not defined, feature added to first view for this seq

seqref	reference sequence
alignment	alignment pos on global seq
startpos	optional start of seq range
endpos	optional end of seq range
refs	space-separated list of IDs

Motif-element Element

Describes one element of motif.

Content Model

<!**ELEMENT** Motif-element **EMPTY**>

Attributes

```
<!ATTLIST Motif-element
  %attrs;
  display-auto      %yesorno; #IMPLIED
  border-color      %color;    #IMPLIED
  fill-fg-color      %color;    #IMPLIED
  fill-bg-color      %color;    #IMPLIED
  fill-pattern       (clear|horiz|vert|fdiag|bdiag|
                     cross|diagcross|solid) #IMPLIED
  column-width       %length;   #IMPLIED
  show-values        %yesorno;  #IMPLIED
  show-titles        %yesorno;  #IMPLIED
  show-no-position   %yesorno;  #IMPLIED
  state-names        CDATA      #IMPLIED
  value-scale        %integer;  #IMPLIED
  value-min          %integer;  #IMPLIED
  value-max          %integer;  #IMPLIED>
```

*Notes

<u>Attribute</u>	<u>Explanation</u>
display-auto	if display-auto, create default display widget and show
border-color	color used to paint border of motif rectangle
-- fill	applies to intervals --
fill-fg-color	foreground of fill area
fill-bg-color	background of fill area
fill-pattern	sets pattern for filling motif rectangle
column-width	defines width of column to display title or value of motif element as nonpositional data to left of sequence line
show-values	if 1, show data values when motif members shown as points or when member is nonpositional
show-titles	if 1, show titles when motif members shown as points or when member is nonpositional
show-no-position	if 1, show as nonpositional information to left of sequence line

state-names	comma-separated list of titles for position values= -1,-2, etc. When members shown as text, state name shown in place of point of interval
value-scale	used to scale integer values for display; divide by power of 10
value-min	to show as saturation (lower limit)
value-max	to show as saturation (upper limit)

Motif-table Element

A **Motif-table** contains hierarchical data for motif matches on any number of sequences.

Content Model

```
<!ELEMENT Motif-table (Attribute*, Numbering?,
                        Interval-loc*, Motif-element*,
                        Sequence-motif*, Resource*[3], %links;) >
```

Attributes

```
<!ATTLIST Motif-table
  %attrs;
  display-height      %length;   #IMPLIED
  display-gap         %length;   #IMPLIED
  point-width         %length;   #IMPLIED
  title-width         %length;   #IMPLIED
  value-width         %length;   #IMPLIED
  values-as-color     %yesorno;  #IMPLIED
  value-scale         %integer;  #IMPLIED
  global-length       %integer;  #IMPLIED
  refs                IDREFS     #IMPLIED>
```

*Notes

<u>Attribute</u>	<u>Explanation</u>
display-height	sets height of interval rectangles on sequence lines of display
display-gap	sets distance between display rectangle and start of next sequence line
point-width	line width for plotting points
title-width	if defined, sets width of area to show titles
value-width	if defined, sets width of area to show values
values-as-color	set to 1 to show values as saturation
value-scale	to scale integer values; divide by power of 10
global-length	=longest seq otherwise
refs	element cross-references

Alignment-point-set Element

An **Alignment-point-set** describes the alignment points of a set of sequences.

Content Model

```
<!ELEMENT Alignment-point-set (Attribute*, Numbering?,
    Resource*[3], %links;) >
```

Attributes

```
<!ATTLIST Alignment-point-set
    %attrs;
    seqids      IDREFS      #REQUIRED
    featids     CDATA       #REQUIRED
    captions    CDATA       #IMPLIED
    consensus   %yesorno;   #IMPLIED
    numseg      %integer;   #IMPLIED
    starts      CDATA       #IMPLIED
    strands     CDATA       #IMPLIED
    seglens     CDATA       #IMPLIED
    refs        IDREFS      #IMPLIED>
```

*Notes

<u>Attribute</u>	<u>Explanation</u>
seqids	ids of sequences
featids	ids of corresponding features, points or intervals, or actual position values; value="NONE" means not included
captions	n - 1 comma-separated captions to be shown on lines connecting points
consensus	true if 1st=consensus seq
-- next 4 borrowed attributes from Dense-seg in ncbi asn.1	--
numseg	number of segments
starts	offsets
strands	"P" or "M" - strand
seglens	lengths of segments
refs	element cross-references

Seq-pair-alignment Element

The regions of similarity between two sequences are described using **Seq-pair-alignment** elements.

Content Model

```
<!ELEMENT Seq-pair-alignment (Attribute*, Seq-pair-run*,
    Resource*[3], %links;)>
```

Attributes

```
<!ATTLIST Seq-pair-alignment
    %attrs;
    reftype     CDATA       #IMPLIED
    refsource    CDATA       #IMPLIED
    refseq       IDREF       #REQUIRED
    refxref      CDATA       #IMPLIED
    refcaption   CDATA       #IMPLIED
    refstart     %integer;   #IMPLIED
```

refend	%integer;	#IMPLIED
reflength	%integer;	#IMPLIED
comptype	CDATA	#IMPLIED
compsource	CDATA	#IMPLIED
compseq	IDREF	#IMPLIED
compxref	CDATA	#IMPLIED
compcaption	CDATA	#IMPLIED
compstart	%integer;	#IMPLIED
compend	%integer;	#IMPLIED
complength	%integer;	#IMPLIED
method	CDATA	#IMPLIED
runminscore	CDATA	#IMPLIED
runmaxscore	CDATA	#IMPLIED
totalscore	CDATA	#IMPLIED
refs	IDREFS	#IMPLIED>

*Notes

<u>Attribute</u>	<u>Explanation</u>
reftype	type of sequence
refsource	source
refseq	reference sequence
refxref	database reference
refcaption	display caption
refstart	overall region searched
refend	end of search region
reflength	complete seq length
comptype	type of sequence
compsource	source
compseq	comparison sequence
compxref	database reference
compcaption	display caption
compstart	overall region searched
compend	end of search region
complength	complete seq length
method	comparison notes
runminscore	minimum run score
runmaxscore	maximum run score
totalscore	total alignment score
refs	element cross-references

Seq-pair-run Element

Each region of similarity between two sequences is described using the **Seq-pair-run** element.

Content Model

```
<!ELEMENT Seq-pair-run (Attribute*, Resource*[3], %links;)>
```

Attributes

```
<!ATTLIST Seq-pair-run
  %attrs;
  id          IDREF          #IMPLIED
  translated   %yesorno;    #IMPLIED
```

runlength	%integer;	#REQUIRED
comprunlength	%integer;	#IMPLIED
refpos	%integer;	#REQUIRED
refcomplement	%yesorno;	#IMPLIED
refdata	CDATA	#IMPLIED
comppos	%integer;	#REQUIRED
compcomplement	%integer;	#IMPLIED
compdata	CDATA	#IMPLIED
runscore	CDATA	#IMPLIED
runprob	CDATA	#IMPLIED
alignment	CDATA	#IMPLIED
refs	IDREFS	#IMPLIED>

*Notes

<u>Attribute</u>	<u>Explanation</u>
id	unique identifier
translated	compared translated
runlength	length on ref seq
comprunlength	if <> runlength
refpos	position on ref
refcomplement	true if comp strand
refdata	sequence data
comppos	position on comp
compcomplement	true if comp strand
compdata	sequence data
runscore	displayable value
runprob	probability
alignment	transform compseq

* The alignment transforms the reference sequence into the comparison sequence; it uses the following terminology: =N next N bases/residues are identical; -N:Mcc...replace next N bases/residues with following M For example =5-3:2ac-0:4acgt=20-4:0

Sequence-search-table Element [3]

The **Sequence-search-table** element refers to sequence search results represented in an external format stored separately (**url**) or in a **Sequence-search-output** element.

Content Model

```
<!ELEMENT Sequence-search-table (Attribute*, Sequence-search-output?, Resource*,%links;)> [3]
```

Attributes

```
<!ATTLIST Sequence-search-table
  %attrs;
  search-type      CDATA      #IMPLIED
  url              %url;      #IMPLIED
  analysis-title   CDATA      #IMPLIED
  analysisref      IDREF      #IMPLIED
  queryseqref      IDREF      #IMPLIED
  query-start      %integer;   #IMPLIED
  query-length     %integer;   #IMPLIED
  on-complement    %yesorno;   #IMPLIED
```

translated	%yesorno;	#IMPLIED
frames	CDATA	#IMPLIED
trans-table	CDATA	#IMPLIED
query-data	CDATA	#IMPLIED>

Sequence-search-output Element [3]

The **Sequence-search-output** element is used for search results (often in CDATA).

Content Model

```
<!ELEMENT Sequence-search-output (#PCDATA) > [3]
```

Multiple-alignment-table Element [3]

The **Multiple-alignment-table** contains alignment data and encapsulates the results of a CLUSTAL analysis.

Content Model

```
<!ELEMENT Multiple-alignment-table (Attribute*, CLUSTAL-parameters?,
Alignment-summary?, Pairwise-alignments?, Alignment-groups?,
Sequence-alignment*, Resource*, %links;)> [3]
```

Attributes

```
<!ATTLIST Multiple-alignment
%attrs;
molecule-type (nucleotide|protein) #REQUIRED>
```

CLUSTAL-parameters Element [3]

Input parameters for a CLUSTALW analysis.

Content Model

```
<!ELEMENT CLUSTAL-parameters EMPTY> [3]
```

Attributes

```
<!ATTLIST CLUSTAL-parameters
version CDATA #IMPLIED
method CDATA #IMPLIED
output CDATA #IMPLIED
outorder CDATA #IMPLIED
ktuple %integer; #IMPLIED
window %integer; #IMPLIED
score CDATA #IMPLIED
topdiags %integer; #IMPLIED
pairgap %integer; #IMPLIED
matrix CDATA #IMPLIED
gapopen %integer; #IMPLIED
endgaps %integer; #IMPLIED
gapext %integer; #IMPLIED
gapdist %integer; #IMPLIED>
```

Alignment-summary Element [3]

The **Alignment-summary** contains a description of each participating sequence.

Content Model

```
<!ELEMENT Alignment-summary (Aligned-sequence*)> [3]
```

Attributes

```
<!ATTLIST Alignment-summary
  seq-type   (nucleotide|protein) #REQUIRED
  seq-format   CDATA               #REQUIRED>
```

Aligned-sequence Element [3]

Each participating sequence is defined by an **Alignment-sequence** element, which may refer to a **Sequence** using the **seqref** attribute.

Content Model

```
<!ELEMENT Aligned-sequence EMPTY> [3]
```

Attributes

```
<!ATTLIST Aligned-sequence
  seqref      IDREF      #IMPLIED
  start       %integer;   #IMPLIED
  on-complement %yesorno; #IMPLIED
  translated   %yesorno;   #IMPLIED
  frame       %integer;   #IMPLIED
  trans-table  CDATA      #IMPLIED
  seqnum      %integer;   #REQUIRED
  name        CDATA      #REQUIRED
  length      %integer;   #REQUIRED>
```

Pairwise-alignments Element [3]

The **Pairwise-alignments** element encapsulates the comparisons of each pair of sequences.

Content Model

```
<!ELEMENT Pairwise-alignments (Aligned-pair*)> [3]
```

Aligned-pair Element [3]

A description of the comparison of a pair, identified by an ordinal number.

Content Model

```
<!ELEMENT Aligned-pair EMPTY> [3]
```


Attributes

```
<!ATTLIST Aligned-pair
  seqnum1    %integer;    #REQUIRED
  seqnum2    %integer;    #REQUIRED
  score      %integer;    #REQUIRED>
```

Aligned-groups Element [3]

A container for each group of sequences.

Content Model

```
<!ELEMENT Aligned-groups (Aligned-group*,Total-alignment)> [3]
```

Attributes

```
<!ATTLIST Aligned-groups
  groups      %integer;    #REQUIRED>
```

Aligned-group Element [3]

Description of a subset of sequences.

Content Model

```
<!ELEMENT Aligned-group EMPTY> [3]
```

Attributes

```
<!ATTLIST Aligned-group
  group        %integer;    #REQUIRED;
  total-seq    %integer;    #REQUIRED;
  group-score  %integer;    #REQUIRED>
```

Total-alignment Element [3]

A measure of the overall alignment for the sequence set.

Content Model

```
<!ELEMENT Total-alignment EMPTY> [3]
```

Attributes

```
<!ATTLIST Total-alignment
  align-score  %integer;    #REQUIRED>
```

Sequence-alignment Element [3]

A container for the results of the alignment for each sequence.

Content Model

```
<!ELEMENT Sequence-alignment (Sequence-data*,Alignment-consensus)> [3]
```

Attributes

```
<!ATTLIST Sequence-alignment
  sequences %integer;    #REQUIRED>
```

Sequence-data Element [3]

The alignment for a sequence.

Content Model

```
<!ELEMENT Sequence-data (#PCDATA)> [3]
```

Attributes

```
<!ATTLIST Sequence-data
  seq-name  CDATA #REQUIRED>
```

Alignment-consensus Element [3]

Consensus for the aligned set.

Content Model

```
<!ELEMENT Alignment-consensus (#PCDATA)> [3]
```

PCR-primer-table Element [3]

The **PCR-primer-table** element contains any number of **PCR-primer** elements and is a representation of the PRIMER3 analysis.

Content Model

```
<!ELEMENT PCR-primer-table (Attribute*, PCR-primer*, PCR-summary?,
  PCR-statistical-analysis?, Resource*, %links;)> [3]
```

PCR-primer Element [3]

Representation for one primer pair.

Content Model

```
<!ELEMENT PCR-primer (Primer-sequence*, PCR-product?) > [3]
```

Attributes

```
<!ATTLIST PCR-primer
  %attrs;
  seqref IDREF #IMPLIED>
```

Primer-sequence Element [3]

Each **Primer-sequence** element represents either a left (**primer-left="1"**) or right (**primer-left="0"**) primer sequence.

Content Model

```
<!ELEMENT Primer-sequence EMPTY> [3]
```

Attributes

```
<!ATTLIST Primer-sequence
  %attrs;
  primer-left      %yesorno;   #REQUIRED
  start            %integer;    #REQUIRED
  length           %integer;    #REQUIRED
  seq              CDATA        #REQUIRED
  tm               %real;       #IMPLIED
  gc-percent       %real;       #IMPLIED
  any              %real;       #IMPLIED
  three-end        %real;       #IMPLIED>
```

PCR-summary Element [3]

Summary data for a primer pair.

Content Model

```
<!ELEMENT PCR-summary EMPTY> [3]
```

Attributes

```
<!ATTLIST PCR-summary
  seq-size          %integer;    #REQUIRED
  include-region-size %integer;    #REQUIRED>
```

PCR-product Element [3]

Characteristics of the PCR product.

Content Model

```
<!ELEMENT PCR-product EMPTY> [3]
```

Attributes

```
<!ATTLIST PCR-product
  product-size      %integer;    #REQUIRED
  pair-any-compl    %real;       #IMPLIED
  pair-3-compl      %real;       #IMPLIED>
```

PCR-statistical-analysis Element [3]

Summary of the statistics for each element of the primer pair.

Content Model

```
<!ELEMENT PCR-statistical-analysis (PCR-statistics+) > [3]
```

CR-statistics Element [3]

Statistical summary for the left (**statistics-left="1"**) or right (**statistics-left="0"**) members.

Content Element

```
<!ELEMENT PCR-statistics EMPTY> [3]
```

Attributes

```
<!ATTLIST PCR-statistics
  statistics-left      %yesorno;    #REQUIRED
  considered           %integer;    #IMPLIED
  too-many-ns         %integer;    #IMPLIED
  in-target            %integer;    #IMPLIED
  in-excl-reg          %integer;    #IMPLIED
  bad-gc-percent       %integer;    #IMPLIED
  no-gc-clamp          %integer;    #IMPLIED
  tm-too-low           %integer;    #IMPLIED
  tm-too-high          %integer;    #IMPLIED
  high-any-compl       %integer;    #IMPLIED
  high-3-compl         %integer;    #IMPLIED
  poly-x              %integer;    #IMPLIED
  high-end-stab        %integer;    #IMPLIED
  ok                   %integer;    #IMPLIED>
```

Sets Element

Basic hierarchical structure composed of any grouping of elements by ID using IDREFs.

Content Model

```
<!ELEMENT Sets (Attribute*, Set*, Resource*[3], %links;)>
```

Attributes

```
<!ATTLIST Sets
  %attrs;>
```

Set Element

Content Model

```
<!ELEMENT Set ((Attribute*, List-member*[3], Resource*[3], %links;)>
```

Attributes

```
<!ATTLIST Set
  %attrs;
  idrefs IDREFS #REQUIRED
```

refs IDREFS #IMPLIED>

*Notes

<u>Attribute</u>	<u>Explanation</u>
idrefs	space-separated list of set members
refs	element cross-references

List-member Element [3]

A member of a list.

Content Model

<!ELEMENT List-member EMPTY> [3]

Attributes

```
<!ATTLIST List-member
  title          CDATA          #IMPLIED
  source         CDATA          #IMPLIED
  comment        CDATA          #IMPLIED>
```

Networks Element [3]

Container for a set of network descriptions.

Content Model

<!ELEMENT Networks (Network*) > [3]

Network Element [3]

Representation of a network, which may represent a directed or undirected graph in which the arcs are weighted or not.

Content Model

```
<!ELEMENT Network (Attribute*, Node*, Arc*,
  Network-selection-set*, Resource*, %links;)> [3]
```

Attributes

```
<!ATTLIST Network
  %attrs;
  directed      %yesorno;      #IMPLIED
  weighted      %yesorno;      #IMPLIED>
```

Node Element [3]

Representation of a node in a network.

Content Model

```
<!ELEMENT Node (Attribute*,Resource*,%links;)> [3]
```

Attributes

```
<!ATTLIST Node
  %attrs;
  node-type %url; #IMPLIED>
```

Arc Element [3]

Representation for an arc in a network.

Content Model

```
<!ELEMENT Arc (Attribute*,Resource*,%links;)> [3]
```

Attributes

```
<!ATTLIST Arc
  %attrs;
  connector-type %url; #IMPLIED
  sourcenode IDREF #REQUIRED
  destnode IDREF #REQUIRED
  weight CDATA #IMPLIED>
```

Network-selection-set Element [3]

Representation for a selection of arcs and an nodes in a network.

Content Model

```
<!ELEMENT Network-selection-set EMPTY> [3]
```

Attributes

```
<!ATTLIST Network-selection-set
  %attrs;
  node-default-selection-state %displaystate; #REQUIRED
  nodes-on IDREFS #IMPLIED
  nodes-off IDREFS #IMPLIED
  nodes-gray IDREFS #IMPLIED
  arc-default-selection-state %displaystate; #REQUIRED
  arcs-on IDREFS #REQUIRED
  arcs-off IDREFS #REQUIRED
  arcs-gray IDREFS #IMPLIED
  use-node-display-levels %yesorno; #IMPLIED
  initial-node-display-level %integer; #IMPLIED
  show-node-selection-state %yesorno; #IMPLIED
  show-node-expansion-state %yesorno; #IMPLIED
  set-connector-width-from-weight %yesorno; #IMPLIED
  min-weight %real; #IMPLIED
  max-weight %real; #IMPLIED
  min-width %real; #IMPLIED
```

max-width

%real;

#IMPLIED>

3.5 Research [3]

Research Element [3]

The **Research** section of a BSML document is optional and contains descriptions of searches, queries, analyses, and experiments.

Content Model

```
<!ELEMENT Research (Attribute*,
                    Searches?, Queries?, Analyses?, Experiments? ) > [3]
```

Searches Element [3]

Container for a set of **Search** elements.

Content Model

```
<!ELEMENT Searches (Attribute*, Search*)> [3]
```

Search Element [3]

Generic description of a search.

Content Model

```
<!ELEMENT Search (Attribute*, Search-conditions?, Search-result-map?,
                  Search-result-set*, Resource*, %links;) > [3]
```

Attributes

```
<!ATTLIST Search
    %attrs;
    database      CDATA      #IMPLIED
    url           %url;      #IMPLIED
    searchid      CDATA      #IMPLIED
    total-results %integer;   #IMPLIED
    results-per-set %integer; #IMPLIED>
```

Search-conditions Element [3]

Generic description of conditions of the search.

Content Model

```
<!ELEMENT Search-conditions (#PCDATA)> [3]
```

Search-result-map Element [3]

Description of result output fields.

Content Model

```
<!ELEMENT Search-result-map (Search-result-field-desc*)>
```

Attributes

```
<!ATTLIST Search-result-map
    %attrs;
>
```

Search-result-field-desc Element [3]

Description of a result field.

Content Model

```
<!ELEMENT Search-result-field-desc EMPTY>
```

Attributes

```
<!ATTLIST Search-result-field-desc
    field-number      %integer;    #IMPLIED
    field-name        CDATA        #IMPLIED
    field-type        CDATA        #IMPLIED>
```

Search-result-set Element [3]

Group of search results (one record in each item).

Content Model

```
<!ELEMENT Search-result-set (Search-result-item*)>
```

Attributes

```
<!ATTLIST Search-result-set
    %attrs;
    offset-in-total-result-set %integer; #IMPLIED
    results-this-set          %integer; #IMPLIED>
```

Search-result-item Element [3]

One search result record.

Content Model

```
<!ELEMENT Search-result-item
    (Search-result-field*, (Link | Cross-reference)*)>
```

Attributes

```
<!ATTLIST Search-result-item
    item-number %integer; #IMPLIED
    select      %yesorno; #IMPLIED>
```

Search-result-field Element [3]

One field in a search result set (record).

Content Model

```
<!ELEMENT Search-result-field (#PCDATA)>
```

Attributes

```
<!ATTLIST Search-result-field
    field-number %integer; #IMPLIED
    field-name CDATA #IMPLIED
    field-type CDATA #IMPLIED>
```

Queries Element [3]

Container for a set of **Query** elements.

Content Model

```
<!ELEMENT Queries (Attribute*, Query*)> [3]
```

Query Element [3]

Generic description of a query.

Content Model

```
<!ELEMENT Query (Attribute*, Query-request*, Query-return*,
    Resource*, %links;)> [3]
```

Attributes

```
<!ATTLIST Query
    %attrs;
    description CDATA #IMPLIED
    database CDATA #IMPLIED
    url %url; #IMPLIED>
```

Query-request Element [3]

One request element content description (represented in CDATA).

Content Model

```
<!ELEMENT Query-request (#PCDATA) > [3]
```

Attributes

```
<!ATTLIST Query-request
    %attrs;
    request-desc CDATA #IMPLIED
    request-type CDATA #IMPLIED>
```

Query-return Element [3]

Generic description of the results returned by a query.

Content Model

```
<!ELEMENT Query-return (#PCDATA) > [3]
```

Attributes

```
<!ATTLIST Query-return
    %attrs;
    index          CDATA          #IMPLIED
    return-desc    CDATA          #IMPLIED
    return-type    CDATA          #IMPLIED
    hit-count      %integer;      #IMPLIED>
```

Analyses Element [3]

Container for a set of **Analysis** elements.

Content Model

```
<!ELEMENT Analyses (Attribute*,Analysis*)> [3]
```

Analysis Element [3]

Generic description of an analysis.

Content Model

```
<!ELEMENT Analysis (Attribute*, Parameter*, Input-data*,
                    Resource*, %links;)> [3]
```

Attributes

```
<!ATTLIST Analysis
    %attrs;
    url      %url; #IMPLIED>
```

Parameter Element [3]

Analysis parameter description.

Content Model

```
<!ELEMENT Parameter EMPTY> [3]
```

Attributes

```
<!ATTLIST Parameter
    name          CDATA          #REQUIRED
    value         CDATA          #REQUIRED
    type          (int|float|string) #IMPLIED
```

url %url; #IMPLIED>

Input-data Element [3]

Generic description of input-data.

Content Model

```
<!ELEMENT Input-data (#PCDATA) > [3]
```

Experiments Element [3]

Container for a set of **Experiment** elements.

Content Model

```
<!ELEMENT Experiments (Attribute*,Experiment*)> [3]
```

Experiment Element [3]

Generic description of an experiment, including protocol description.

Content Model

```
<!ELEMENT Experiment (Attribute*, Description?,  
  (Enzyme | Equipment | Materials | Note | Reagent |  
  Product | Sample | Solution | Structure | Supplier | Terminology)*,  
  Protocol*, Resource*, %links;)> [3]
```

Attributes

```
<!ATTLIST Experiment  
  %attrs;  
  url    %url; #IMPLIED>
```

Measure-quantity Element [3]

Description of the quantity.

Content Model

```
<!ELEMENT Measure-quantity (#PCDATA) >
```

Measure-units Element [3]

Description of the units of measurement.

Content Model

```
<!ELEMENT Measure-units (#PCDATA) > [3]
```

Measure Element [3]

Description of a measured quantity in terms of the value and units.

Content Model

```
<!ELEMENT Measure (Measure-quantity,Measure-units?) > [3]
```

Measure-range Element [3]

Description of a region relative to one measured quantity or between two measured quantities.

Content Model

```
<!ELEMENT Measure-range (Measure+) > [3]
```

Attributes

```
<!ATTLIST Measure-range
  range-type (lt|lte|gt|gte|between-exclusive|between-inclusive)
             "between-inclusive">
```

Log-entry Element [3]

Description of actions taken in completing a protocol.

Content Model

```
<!ELEMENT Log-entry (#PCDATA | Measure | Measure-range)* > [3]
```

Attributes

```
<!ATTLIST Log-entry
  id                ID                #IMPLIED
  event             CDATA             #IMPLIED
  event-time        %datetime;        #IMPLIED
  url               %url;             #IMPLIED>
```

Description Element [3]

Generic element used for descriptions.

Content Model

```
<!ELEMENT Description (#PCDATA | Log-entry | Measure |
                      Measure-range | Link)* > [3]
```

Note Element [3]

Description of a precaution or warning, research note, etc.

Content Model

```
<!ELEMENT Note (Attribute*, Description?, Resource*, %links;)> [3]
```

Attributes

```
<!ATTLIST Note
  %attrs;
  url          %url;          #IMPLIED>
```

Enzyme Element [3]

Description of an enzyme. The **title** indicates the name of the enzyme, and the **value-type** attribute is used to describe the type of enzyme (restriction, etc.).

Content Model

```
<!ELEMENT Enzyme (Attribute*, Description?, Resource*, %links;)> [3]
```

Attributes

```
<!ATTLIST Enzyme
  %attrs;
  url          %url;          #IMPLIED
  ec-number     CDATA         #IMPLIED
  recognition-sequence CDATA   #IMPLIED
  offset        %integer;     #IMPLIED
  overhang       %integer;     #IMPLIED>
```

Equipment Element [3]

Description of equipment.

Content Model

```
<!ELEMENT Equipment (Attribute*, Description?, Resource*, %links;)> [3]
```

Attributes

```
<!ATTLIST Equipment
  %attrs;
  url %url; #IMPLIED>
```

Materials Element [3]

Description of materials.

Content Model

```
<!ELEMENT Materials (Attribute*, Description?, Resource*, %links;)> [3]
```

Attributes

```
<!ATTLIST Materials
  %attrs;
```

```
url %url; #IMPLIED>
```

Product Element [3]

Description of an experimental product that may be the output of one step and/or the input to another step.

Content Model

```
<!ELEMENT Product (Attribute*, Description?, Resource*, %links;)> [3]
```

Attributes

```
<!ATTLIST Product
  %attrs;
  url %url; #IMPLIED>
```

Reagent Element [3]

Description of a reagent.

Content Model

```
<!ELEMENT Reagent (Attribute*, Description?, Resource*, %links;)> [3]
```

Attributes

```
<!ATTLIST Reagent
  %attrs;
  url %url; #IMPLIED>
```

Sample Element [3]

Description of a sample.

Content Model

```
<!ELEMENT Sample (Attribute*, Description?,
  (Organism | Organism-clone | Cell-line)?,
  Development-stage?, Tissue-sample*,
  Resource*, %links;)> [3]
```

Attributes

```
<!ATTLIST Sample
  %attrs;
  url %url; #IMPLIED>
```

Organism Element [3]

Description of an organism.

Content Model

```
<!ELEMENT Organism (Attribute*, Description? Strain*,
                    Resource*, %links;)> [3]
```

Attributes

```
<!ATTLIST Organism
  %attrs;
  genus      CDATA #IMPLIED
  species     CDATA #IMPLIED
  strain      CDATA #IMPLIED
  taxon-num   CDATA #IMPLIED
  taxonomy    CDATA #IMPLIED
  url         %url; #IMPLIED>
```

Strain Element [3]

Content Model

```
<!ELEMENT Strain (Attribute*, Description?,
                  (Organism-clone | Cell-line)*,
                  Resource*, %links;)>
```

Attributes

```
<!ATTLIST Strain
  %attrs;
  url %url; #IMPLIED>
```

Organism-clone Element [3]

Description of an organism clone.

Content Model

```
<!ELEMENT Organism-clone (Attribute*, Description?,
                           Resource*, %links;)> [3]
```

Attributes

```
<!ATTLIST Organism-clone
  %attrs;
  url %url; #IMPLIED>
```

Cell-line Element [3]

Description of a cell line.

Content Model

```
<!ELEMENT Cell-line (Attribute*, Description?, Resource*, %links;) >
```

Attributes

```
<!ATTLIST Cell-line
  %attrs;
```


url %url; #IMPLIED>

Development-stage Element [3]

Description of a developmental stage.

Content Model

<!ELEMENT Development-stage (#PCDATA)> [3]

Attributes

```
<!ATTLIST Development-stage
  id                ID                #IMPLIED
  name              CDATA            #REQUIRED
  stage-age-value   CDATA            #IMPLIED
  stage-age-units   %timeunits;      #IMPLIED>
```

Tissue-sample Element [3]

Description of a tissue sample.

Content Model

<!ELEMENT Tissue-sample (#PCDATA) > [3]

Attributes

```
<!ATTLIST Tissue-sample
  id                ID                #IMPLIED
  tissue-type       CDATA            #IMPLIED
  organ            CDATA            #IMPLIED
  tissue-age-value  CDATA            #IMPLIED
  tissue-age-units  %timeunits;      #IMPLIED>
```

Solution Element [3]

Description of a solution.

Content Model

<!ELEMENT Solution (Attribute*, Description?, Resource*, %links;)> [3]

Attributes

```
<!ATTLIST Solution
  %attrs;
  url          %url;          #IMPLIED>
```

Structure Element [3]

Description of a relevant structure (e.g., protein structure).

Content Model

```
<!ELEMENT Structure (Attribute*, Description?, Resource*, %links;)> [3]
```

Attributes

```
<!ATTLIST Structure
  %attrs;
  url      %url;      #IMPLIED>
```

Supplier Element [3]

Description of the supplier of reagents, equipment, etc. **Resource** elements may be used to associate organization names and addresses with a supplier.

Content Model

```
<!ELEMENT Supplier (Attribute*, Description?, Resource*, %links;)> [3]
```

Attributes

```
<!ATTLIST Supplier
  %attrs;
  url      %url;      #IMPLIED>
```

Terminology Element [3]

Description of a term or glossary entry.

Content Model

```
<!ELEMENT Terminology (Attribute*, Description?,
                        Resource*, %links;)> [3]
```

Attributes

```
<!ATTLIST Terminology
  %attrs;
  url      %url;      #IMPLIED>
```

Protocol Element [3]

Description of a protocol.

Content Model

```
<!ELEMENT Protocol (Attribute*, Description?,
                   Protocol-step*, Resource*, %links;)> [3]
```

Attributes

```
<!ATTLIST Protocol
  %attrs;
  url      %url;      #IMPLIED>
```

Protocol-step Element [3]

Description of a step in the protocol.

Content Model

```
<!ELEMENT Protocol-step (Attribute*, Description*,  
                          Resource*, %links;)> [3]
```

Attributes

```
<!ATTLIST Protocol-step  
  %attrs;  
  url %url; #IMPLIED>
```

3.6 Display

Display Element

The **Display** element is a container for the optional display widgets and display content.

Content Model

```
<!ELEMENT Display (Links?,Fonts?,Styles?,Screen?,Paper?,Page+)>
```

Attributes

```
<!ATTLIST Display
  master-doc      %yesorno;      #IMPLIED
  first-page      %integer;       #IMPLIED
  default-unit    (pct|cm|in|px) "in"
  line-width-base %length;       "0.01in"
  line-width-inc  %length;       "0.01in">
```

Notes*

<u>Attribute</u>	<u>Explanation</u>
master-doc	specifies whether current document should be regarded as master document (=1) or not (=0)
first-page	display 1 st when document is opened
default-unit	specifies unit for distance or length attribute
line-width-base	specifies base width for line widths specified
line-width-inc	specifies increment to use in computing line widths

* screen and paper set defaults for all pages, if defined

Fonts Element

BSML uses a modified version of the HTML 4.0 font element. Fonts are described by their face, color, size (pts), and style (bold,italics,underline). Fonts may refer to base fonts and specify only attributes that they wish to change. Display objects refer to a specific font or use default fonts assigned for particular types of text.

The **Fonts** element defines defaults and provides a container for each defined **Font**.

Content Model

```
<!ELEMENT Fonts (Font*)>
```

Attributes

```
<!ATTLIST Fonts
  base-font  IDREF #IMPLIED
  text-font  IDREF #IMPLIED
  numb-font  IDREF #IMPLIED
  fixed-font IDREF #IMPLIED
  font-sizes CDATA #IMPLIED>
```

Notes

<u>Attribute</u>	<u>Explanation</u>
base-font	points to main default font
text-font	points to text default font
numb-font	points to numbering default font
fixed-font	points to fixed width default font
font-sizes	comma-separated list of sizes specified in points for font size 1-7

Fonts Element

The Font element is adapted from the html 4.0 specification, but is not used as an inline formatting element.

Content Model

<!ELEMENT Font EMPTY>

Attributes

```
<!ATTLIST Font
  id          ID          #IMPLIED
  class       CDATA       #IMPLIED
  title       CDATA       #IMPLIED
  comment     CDATA       #IMPLIED
  reffont     IDREF       #IMPLIED
  size        CDATA       #IMPLIED
  color       CDATA       #IMPLIED
  face        CDATA       #IMPLIED
  bold        %yesorno;   #IMPLIED
  italic      %yesorno;   #IMPLIED
  underline   %yesorno;   #IMPLIED
  fixed       %yesorno;   "0">
```

Notes

<u>Attribute</u>	<u>Explanation</u>
id	document-wide unique id for selecting font
class	space-separated list of classes
title	displayable title
comment	added for BSML
reffont	reference font for inheritance - if no reference font is given, the default font is used.
size	sets the size of the font. Possible values: a. An integer between 1 and 7. This sets the font to a fixed size; rendering depends on the user agent. Not all user agents may render all seven sizes. b. A relative increase in font size. The value "+1" means one size larger. The value "-3" means three sizes smaller.

All sizes belong to the scale of 1 to 7.

c. An absolute size in pts, e.g., size="14pt".

[+]nn e.g. size="+1", size=4

sets the text color.

#RRGGBB in hex, e.g., red: "#FF0000"

defines a comma-separated list of font names

the user agent should search for in order of preference.

1 = bold, 0 = normal (BSML)

1 = italic, 0 = normal (BSML)

1 = underline, 0 = normal (BSML)

set as fixed pitch font (BSML)

Styles

Styles Element

BSML allows the specification of default values based upon Cascading Style Sheets (CSS), which are contained in the **Styles** element.

Content Model

```
<!ELEMENT Styles (Style-import*,Style?)>
```

Style-import Element

Imported CSS file.

Content Model

```
<!ELEMENT Style-import EMPTY>
```

Attributes

```
<!ATTLIST Style-import
  %attrs;
  source      %url;  #REQUIRED
```

Notes

<u>Attribute</u>	<u>Explanation</u>
source	CSS document

Style Element

CSS style information contained in the document.

Content Model

```
<!ELEMENT Style (#PCDATA)>
```

Attributes

```
<!ATTLIST Style
  type      %contenttype; #REQUIRED
  media     %mediadesc;   #IMPLIED
  title     %text;        #IMPLIED>
```

Notes

<u>Attribute</u>	<u>Explanation</u>
type	content type of style lang. = "css/text" for now
media	for use with these media
title	advisory title

Pages, Views, and Display Objects

Object Element

A generic embedded object that may be used to control visualization.

Content Model

```
<!ELEMENT Object (Param)*>
```

Attributes

```
<!ATTLIST Object
  %attrs;
  declare      (declare) #IMPLIED
  classid     %url;      #IMPLIED
  codebase    %url;      #IMPLIED
  data        %url;      #IMPLIED
  type        %contenttype; #IMPLIED
  codetype    %contenttype; #IMPLIED
  archive     %url;      #IMPLIED
  standby     %text;     #IMPLIED
  height      %length;   #IMPLIED
  width       %length;   #IMPLIED
  usemap      %url;      #IMPLIED
  shapes      (shapes)   #IMPLIED
  export      (export)   #IMPLIED
  name        CDATA      #IMPLIED
  tabindex    %integer;  #IMPLIED>
```

Notes

<u>Attribute</u>	<u>Explanation</u>
declare	declare but don't instantiate flag
classid	identifies an implementation
codebase	base URL for classid, data, archive
data	reference to object's data
type	content type for data
codetype	content type for code

archive	space-separated archive list
standby	message to show while loading
height	override height
width	override width
usemap	use client-side image map
shapes	object has shaped hypertext links
export	export shapes to parent
name	submit as part of form
tabindex	position in tabbing order

Param Element

Named property value used by the **Object** element.

Content Model

```
<!ELEMENT Param EMPTY>
```

Attributes

```
<!ATTLIST Param
  id          ID          #IMPLIED
  name        CDATA      #REQUIRED
  value       CDATA      #IMPLIED
  valuetype   (data|ref|object) "data"
  type        %contenttype; #IMPLIED>
```

Notes

<u>Attribute</u>	<u>Explanation</u>
id	document-wide unique id
name	property name
value	property value
valuetype	how to interpret value
type	content type for value when valuetype=ref

Image-map Element

Client-side image map used by the **File-widget** element.

Content Model

```
<!ELEMENT Image-map (Area)+>
```

Attributes

```
<!ATTLIST Image-map
  %attrs;
  name          CDATA      #REQUIRED>
```

Notes

<u>Attribute</u>	<u>Explanation</u>
name	name of image map for refs

Area Element

Client-side image map area .

Content Model

```
<!ELEMENT Area (Attribute*,Resource*,%links;) >
```

Attributes

```
<!ATTLIST Area
  %attrs;
  shape      %shape;      "rect"
  coords     %coords;     #IMPLIED
  href       %url;        #IMPLIED
  target     %frametarget; #IMPLIED
  nohref     (nohref)     #IMPLIED
  alt        %text;       #REQUIRED
  tabindex   %integer;    #IMPLIED
  accesskey  %character;  #IMPLIED
  onfocus   %script;     #IMPLIED
  onblur     %script;     #IMPLIED>
```

Notes*

<u>Attribute</u>	<u>Explanation</u>
shape	interpretation of coords
coords	comma-separated length list
href	URL for linked resource
target	render in this frame
nohref	this region has no action
alt	alt
tabindex	tabindex
accesskey	accessibility key character
onfocus	the element got the focus
onblur	the element lost the focus

Page Utility Elements

Coord Element

Coordinates of a point on the page.

Content Model

```
<!ELEMENT Coord EMPTY >
```

Attributes

```
<!ATTLIST Coord
  %attrs;
  hloc %length; #IMPLIED
  vloc %length; #IMPLIED>
```

Notes

<u>Attribute</u>	<u>Explanation</u>
hloc	horizontal coordinate
vloc	vertical coordinate

Quantifier Element

Display aspect quantifier.

Content Model

```
<!ELEMENT Quantifier EMPTY>
```

Attributes

```
<!ATTLIST Quantifier
  %attrs;
  type (color|saturation|length) #REQUIRED
  min-val %real; "0.0"
  max-val %real; "1.0"
  val-name CDATA #IMPLIED
  base %real; #IMPLIED
  slope %real; #IMPLIED
  minimum %real; #IMPLIED
  maximum %real; #IMPLIED
  red-base %real; #IMPLIED
  red-slope %real; #IMPLIED
  blue-base %real; #IMPLIED
  blue-slope %real; #IMPLIED
  green-base %real; #IMPLIED
  green-slope %real; #IMPLIED
  to-white %yesorno; #IMPLIED>
```

Notes

<u>Attribute</u>	<u>Explanation</u>
type	specifies the line characteristic to be controlled
min-val	(value - min)/(max-min)
max-val	
val-name	if color or length
-- base and slope used if not color --	
base	$y = \text{slope} * \text{val} + \text{base}$
slope	
minimum	do not exceed for output values
maximum	do not exceed for output values
red-base	if type=color
blue-base	if type=color
green-base	if type=color

to-white

to-white

Margin Element

Used to set boundaries of a page or display region.

Content Model

```
<!ELEMENT Margin EMPTY >
```

Attributes

```
<!ATTLIST Margin
  %attrs;
  margin      %length; #IMPLIED
  horizontal  %length; #IMPLIED
  vertical    %length; #IMPLIED
  left        %length; #IMPLIED
  right       %length; #IMPLIED
  top         %length; #IMPLIED
  bottom      %length; #IMPLIED>
```

Screen Element

May be attached to the **Display** element or to a **Page**. Sets default attributes of screen display.

Content Model

```
<!ELEMENT Screen (Margin?)>
```

Attributes

```
<!ATTLIST Screen
  %attrs;
  width      %length; #IMPLIED
  height     %length; #IMPLIED
  color      %color; #IMPLIED
  monochrome %yesorno; "0">
```

Notes

<u>Attribute</u>	<u>Explanation</u>
width	width of drawing canvas
height	height of drawing canvas
color	color for screen, or white
monochrome	print black/white only

Paper Element

May be attached to the **Display** element or to a **Page**. Sets default attributes of printed output.

Content Model

`<!ELEMENT Paper (Margin?)>`

Attributes

```
<!ATTLIST Paper
  %attrs;
  monochrome %yesorno; "0"
  landscape  %yesorno; "0">
```

Notes

<u>Attribute</u>	<u>Explanation</u>
monochrome	print black/white only
landscape	if not, print portrait

Border Element

A border around a rectangular area.

Content Model

`<!ELEMENT Border EMPTY>`

Attributes

```
<!ATTLIST Border
  %attrs;
  line-width %linewidth; #IMPLIED
  line-color %color;     #IMPLIED
  gutter      %length;   #IMPLIED>
```

Notes

<u>Attribute</u>	<u>Explanation</u>
gutter	indent to border

Symbol Element

Symbol display object; defines shape and size of a symbol.

Content Model

`<!ELEMENT Symbol EMPTY>`

Attributes

```
<!ATTLIST Symbol
  %attrs;
  shape (opencircle|closedcircle|
        opensquare|closedsquare|triangle) #IMPLIED
  color %color; #IMPLIED
  diam  %length; #IMPLIED>
```

Notes

<u>Attribute</u>	<u>Explanation</u>
color	use default if unspecified
diam	use default if unspecified

X-axis and Y-axis Element

Used for scaling a sequence or other quantified data.

Content Model

```
<!ELEMENT X-axis|Y-axis (Numbering?)>
```

Attributes

```
<!ATTLIST (X-axis|Y-axis)
  use-interval    %yesorno;    #IMPLIED
  interval-value  %integer;    #IMPLIED
  tick-length     %length;     #IMPLIED
  tick-width      %linewidth;  #IMPLIED
  tick-color      %color;      #IMPLIED
  axis-font       IDREF        #IMPLIED>
```

Notes

<u>Attribute</u>	<u>Explanation</u>
use-interval	else default
interval-value	if user-int
tick-length	scale tick
tick-width	tick width
tick-color	tick mark color
axis-font	font reference

Chart Element

Chart display to locate chart data.

Content Model

```
<!ELEMENT Chart (Border?)>
```

Attributes

```
<!ATTLIST Chart
  %attrs;
  source          %url;        #IMPLIED
  tableid         IDREF        #IMPLIED
  bycolumn        %yesorno;    "1"
  colrownum       %integer;    #IMPLIED>
```

Notes

<u>Attribute</u>	<u>Explanation</u>
source	specifies the url for the table if an external file is used

tableid	specifies unique id of an element in Tables section if table is used
bycolumn	select data by rows if ="2" or by column if ="1"
colrownum	number of row or col from which data are selected

Member Element

Used to define sequence point in a point group.

Content Model

```
<!ELEMENT Member EMPTY>
```

Attributes

```
<!ATTLIST Member
  caption      CDATA      #IMPLIED
  position     %integer;  #REQUIRED
  plus         %yesorno;  #IMPLIED
  display      %yesorno;  #IMPLIED
  value        CDATA      #IMPLIED>
```

Notes

<u>Attribute</u>	<u>Explanation</u>
caption	optional title
position	sequence position
plus	on plus strand
display	select to show
value	optional

Display Set Types

A Simple-set

Simple-set-widget Element

A visualization widget for a simple collection of **Set** elements.

Content Model

```
<!ELEMENT Simple-set-widget (Coord,Border?,Set*,Object?,
                             Resource*, %links;)>
```

Attributes

```
<!ATTLIST Simple-set-widget
  %attrs;
  own-window %yesorno;  #IMPLIED
  setids     IDREFS     #IMPLIED
  caption    CDATA      #IMPLIED
  capt-font  IDREF      #IMPLIED
```

```
list-font    IDREF          #IMPLIED
align       (left|center|right) #IMPLIED
line-link   %yesorno;      #IMPLIED
line-width  %linewidth;    #IMPLIED
line-color  %color;        #IMPLIED
depth       %integer;      #IMPLIED
indent      %length;       #IMPLIED>
```

Notes

<u>Attribute</u>	<u>Explanation</u>
own-window	embed on page if false
setids	ids of set
caption	display title
capt-font	caption font
list-font	item list font
align	align list
line-link	link set members
line-width	linking line
line-color	line color
depth	recursion depth
indent	distance for child

A Tree-set

A tree set is a hierarchy or network displayed as a tree. The basic display includes titles obtained from the list of members using their ids.

Tree-node Element

Represents a node in the tree.

Content Model

```
<!ELEMENT Tree-node EMPTY>
```

Attributes

```
<!ATTLIST Tree-node
  %attrs;
  hloc      %length;      #REQUIRED
  vloc      %length;      #REQUIRED
  setid     IDREF         #IMPLIED
  parentid  IDREF         #IMPLIED
  node-font IDREF         #IMPLIED
  line-width %linewidth;  #IMPLIED
  line-color %color;      #IMPLIED
  branchlen %length;      #IMPLIED
  text-gap  %length;      #IMPLIED
  text-space %length;     #IMPLIED
  node-caption CDATA      #IMPLIED
  node-cap-valign (top|middle|bottom|baseline) #IMPLIED
  node-cap-halign (left|center|right) #IMPLIED
```

branch-caption CDATA	#IMPLIED
branch-cap-above %yesorno;	#IMPLIED>

Notes

<u>Attribute</u>	<u>Explanation</u>
hloc	relative position on page
vloc	relative position on page
setid	if none = node only
parented	if none is root
node-font	item list font
line-width	linking line
line-color	line color
branchlen	branch length
text-gap	gap line to text
text-space	vertical space
node-caption	print at node
node-cap-valign	vertical alignment of node caption
node-cap-halign	horizontal alignment of node caption
branch-cap-above	true=above line

Tree-set-widget Element

Represents the complete tree.

Content Model

```
<!ELEMENT Tree-set-widget (Coord,Border?,Tree-node*,Object?,
                           Resource*,%links;) >
```

Attributes

```
<!ATTLIST Tree-set-widget
  %attrs;
  seqtype          (nucleotide|protein|both) #IMPLIED
  elementtype      (sequence|feature|mixed)  #IMPLIED
  caption          CDATA                     #IMPLIED
  capt-font        IDREF                     #IMPLIED
  node-cap-font     IDREF                     #IMPLIED
  branch-cap-font  IDREF                     #IMPLIED
  leaf-value-font  IDREF                     #IMPLIED
  show-leaf-values %yesorno;                 #IMPLIED
  show-sequence    %yesorno;                 #IMPLIED
  orientation      (vertical|horizontal)     #IMPLIED
  link-straight    %yesorno;                 #IMPLIED
  node-font        IDREF                     #IMPLIED
  line-width       %linewidth;               #IMPLIED
  line-color       %color;                   #IMPLIED
  branchlen        %length;                  #IMPLIED
  text-gap         %length;                  #IMPLIED
  text-space       %length;                  #IMPLIED>
```

Notes

<u>Attribute</u>	<u>Explanation</u>
-- content attributes, sequence includes intervals --	
seqtype	
elementtype	
-- display attributes --	
caption	display title
capt-font	caption font
node-cap-font	node captions
branch-cap-font	branch captions
leaf-value-font	leaf captions
show-leaf-values	show if true
show-sequence	only names if no
orientation	sequences
link-straight	straight lines
	if link-straight is false, rectangular links drawn
-- attributes used by default for all links --	
node-font	item list font
line-width	linking line
line-color	line color
branchlen	branch length
text-gap	gap line to text
text-space	vertical spacing

Alignment-point-sets

Alignment-point-sets-widget Element

This element provides graphic visualization of sequences and points of alignment linking one member of the set to another.

Content Model

```
<!ELEMENT Alignment-point-sets-widget (Symbol?,
    Alignment-point-set*,Object?, Resource*, %links;)>
```

Attributes

```
<!ATTLIST Alignment-point-sets-widget
    %attrs;
    pointsets IDREFS #REQUIRED
    alignto IDREF #IMPLIED
    viewids IDREFS #REQUIRED
    capt-font IDREF #IMPLIED
    capt-above %yesorno; #IMPLIED
    line-width %length; #IMPLIED
    line-color %color; #IMPLIED>
```

Notes

<u>Attribute</u>	<u>Explanation</u>
------------------	--------------------

pointsets	ids of point sets
alignto	reference view
viewids	ids of views
capt-font	caption font
capt-above	captions above line

Dot-plot using Seq-pair-alignment

Dot-plot-widget Element

Display a dot matrix plot for a pair of sequences showing runs of alignment.

Content Model

```
<!ELEMENT Dot-plot-widget (Coord,
                           X-axis?,Y-axis?,Quantifier?,Object?,
                           Resource*,%links;) >
```

Attributes

```
<!ATTLIST Dot-plot-widget
  %attrs;
  seqpair          IDREF          #REQUIRED
  width            %length;       #IMPLIED
  height           %length;       #IMPLIED
  use-x-axis       %yesorno;      #IMPLIED
  use-y-axis       %yesorno;      #IMPLIED
  y-proportional   %yesorno;      #IMPLIED
  y-on-top         %yesorno;      #IMPLIED
  line-width       %linewidth;    #IMPLIED
  line-color       %color;        #IMPLIED
  border-width     %linewidth;    #IMPLIED
  border-color     %color;        #IMPLIED
  min-runlength    %integer;      #IMPLIED
  min-runscore     %real;         #IMPLIED
  max-runscore     %real;         #IMPLIED>
```

Notes

<u>Attribute</u>	<u>Explanation</u>
seqpair	reference pair align
y-proportional	ignore height
y-on-top	number from top down
min-runlength	display threshold
min-runscore	display threshold
max-runscore	display threshold

Network Representation Linking Elements [3]

Representation of directed and undirected graphs to be used to capture semantics of pathways, etc., using **Links** to refer to underlying biological objects.

Network-display-widget Element [3]

Display element for a network.

Content Model

```
<!ELEMENT Network-display-widget (Coord?, Node-display*,
                                   Arc-display*,Object?, Resource*, %links;)> [3]
```

Attributes

```
<!ATTLIST Network-display-widget
  %attrs;
    networkref          IDREF          #REQUIRED
    canvascolor         %color;        #IMPLIED
    nodetextfont         IDREF          #IMPLIED
    nodetextcolor       %color;        #IMPLIED
    nodebgcolor         %color;        #IMPLIED
    nodebordercolor     %color;        #IMPLIED
    nodeborderstrokesize %length;      #IMPLIED
    arctextfont         IDREF          #IMPLIED
    arctextcolor        %color;        #IMPLIED
    arcdashline         %yesorno;      #IMPLIED
    arcline-color       %color;        #IMPLIED
    arcline-width       %length;       #IMPLIED
    arc-diam            %length;       #IMPLIED
    arc-len             %length;       #IMPLIED
    own-window          %yesorno;      #IMPLIED
    width               %length;       #IMPLIED
    height              %length;       #IMPLIED
    draw-border         %yesorno;      #IMPLIED
    scale-pct           %integer;      #IMPLIED
    showgrid            %yesorno;      #IMPLIED
    gridcolor           %color;        #IMPLIED
    gridspacing         %length;       #IMPLIED
    shadow              %yesorno;      #IMPLIED>
```

Node-display Element [3]

Visual component for a node in a network.

Content Model

```
<!ELEMENT Node-display EMPTY> [3]
```

Attributes

```
<!ATTLIST Node-display
    noderef          IDREF          #REQUIRED
    networkref       IDREF          #IMPLIED
    node-icon        %url;          #IMPLIED
    node-image       %url;          #IMPLIED
    node-shape       %displayshape;  #IMPLIED
    text-font-name   CDATA          #IMPLIED
    text-font-style  %integer;      #IMPLIED
    text-font-size   %integer;      #IMPLIED>
```

textcolor	%color;	#IMPLIED
bgcolor	%color;	#IMPLIED
bordercolor	%color;	#IMPLIED
borderstrokesize	%length;	#IMPLIED
left	%length;	#IMPLIED
top	%length;	#IMPLIED
width	%length;	#IMPLIED
height	%length;	#IMPLIED
displaystate	%displaystate;	#IMPLIED
showexpand	%yesorno;	#IMPLIED
expanded	%yesorno;	#IMPLIED>

Arc-display Element [3]

Visual component for an arc in a network.

Content Model

<!ELEMENT Arc-display EMPTY> [3]

Attributes

```
<!ATTLIST Arc-display
  %attrs;
  arcref          IDREF          #IMPLIED
  text-font-name   CDATA          #IMPLIED
  text-font-style  %integer;      #IMPLIED
  text-font-size   %integer;      #IMPLIED
  textcolor        %color;        #IMPLIED
  text-position    %textposition; #IMPLIED
  dashline         %yesorno;      #IMPLIED
  line-mode (straight | orthogonal) "straight"
  line-color       %color;        #IMPLIED
  line-width       %length;       #IMPLIED
  source-point     %direction;    #IMPLIED
  source-arc-type  %connectortype; #IMPLIED
  source-arc-filled %yesorno;     #IMPLIED
  source-arc-diam  %length;       #IMPLIED
  source-arc-len   %length;       #IMPLIED
  dest-point       %direction;    #IMPLIED
  dest-arc-type    %connectortype; #IMPLIED
  dest-arc-filled  %yesorno;     #IMPLIED
  dest-arc-diam    %length;       #IMPLIED
  dest-arc-len     %length;       #IMPLIED
  display-state    %displaystate; #IMPLIED>
```

View (Sequence)-relative Display Widgets

View-line-widget Element

Controls the display of the sequence line.

Content Model

<!ELEMENT View-line-widget EMPTY>

Attributes

```
<!ATTLIST View-line-widget
  %attrs;
  strands          (one|two)      #IMPLIED
  shape            (circular|horizontal|vertical) #IMPLIED
  hcenter          %length;       #IMPLIED
  vcenter          %length;       #IMPLIED
  haligned         %yesorno;      #IMPLIED
  halignview       IDREF          #IMPLIED
  valigned         %yesorno;      #IMPLIED
  valignview       IDREF          #IMPLIED
  width            %linewidth;    #IMPLIED
  gap              %length;       #IMPLIED
  linear-length    %length;       #IMPLIED
  circular-diam    %length;       #IMPLIED
  plus-color       %color;        #IMPLIED
  minus-color      %color;        #IMPLIED
  interval-gap-width %length;     #IMPLIED
  site-criterion   CDATA          #IMPLIED
  group-criterion  CDATA          #IMPLIED
  interval-criterion CDATA        #IMPLIED
  interval-as-point CDATA        #IMPLIED
```

Notes

<u>Attribute</u>	<u>Explanation</u>
hcenter	center - horizontal
vcenter	center - vertical
haligned	aligned to view
halignview	view ref
valigned	aligned to view
valignview	view ref
width	strand line width
gap	between strands
linear-length	line length
circular-diam	circle diam
plus-color	strand color
minus-color	strand color
interval-gap-width	gap for intervals
site-criterion	absolute units or % of complete sequence in current view below which to display
group-criterion	absolute units or % of complete sequence in current view below which to display
interval-criterion	absolute units or % of complete sequence in current view below which to display
interval-as-point	absolute units or % of complete sequence in current view below which to show as point

View-axis-widget Element

Controls location and presentation of numbering information on the sequence line.

Content Model

<!ELEMENT View-axis-widget EMPTY>

Attributes

```
<!ATTLIST View-axis-widget
  %attrs;
  linear-at-side    %yesorno; #IMPLIED
  linear-offset     %length;  #IMPLIED
  circular-rotate   %yesorno; #IMPLIED
  base-at-top       %integer;  #IMPLIED
  use-interval       %yesorno; #IMPLIED
  interval-value     %integer; #IMPLIED
  axis-strand        (both|minus|plus) #IMPLIED
  tick-length        %length;  #IMPLIED
  tick-width         %linewidth; #IMPLIED
  tick-color         %color;    #IMPLIED
  axis-font          IDREF      #IMPLIED>
```

Notes

<u>Attribute</u>	<u>Explanation</u>
linear-at-side	if linear
linear-offset	from seq line
circular-rotate	base at top
base-at-top	if circular
use-interval	else default
interval-value	if user-int
axis-strand	strand
tick-length	scale tick
tick-width	tick-width
tick-color	tick mark color
axis-font	font reference

Point-widget Element

Describes one point on a sequence.

Content Model

<!ELEMENT Point-widget (Symbol?,Object?, Resource*, %links;)>

Attributes

```
<!ATTLIST Point-widget
  %attrs;
  featureref        IDREF      #IMPLIED
  position           %integer;  #IMPLIED
  on-strand          (both|plus|minus|source) #IMPLIED
  caption            CDATA      #IMPLIED
  caption-font       IDREF      #IMPLIED
  position-font      IDREF      #IMPLIED
```

align-text	(auto-align top center bottom)	#IMPLIED
text-length	%length;	#IMPLIED
wrap-text	%yesorno;	#IMPLIED
show-text	%yesorno;	#IMPLIED
show-position	%yesorno;	#IMPLIED
line-gap	%length;	#IMPLIED
text-gap	%length;	#IMPLIED
line-width	%length;	#IMPLIED
line-color	%color;	#IMPLIED
line-length	%length;	#IMPLIED>

Notes

<u>Attribute</u>	<u>Explanation</u>
featureref	id of feature
position	from source feature, if not specified
on-strand	locate on strand if true
caption	no text needed
caption-font	select by idref
position-font	font by idref
align-text	location to align text
text-length	maximum line length
wrap-text	wrap at max line
show-text	display caption
show-position	display pos number
line-gap	gap to seq line
text-gap	gap to caption
line-width	pointer width
line-color	pointer color
line-length	pointer length

Point-group-widget Element

A display element for a set of points on the sequence.

Content Model

```
<!ELEMENT Point-group-widget (Quantifier?,Member*,Object?,
                               Resource*, %links;)>
```

Attributes

```
<!ATTLIST Point-group-widget
  %attrs;
  featureref      IDREF      #IMPLIED
  show-as-one     %yesorno;  #IMPLIED
  check-members   %yesorno;  #IMPLIED
  group-pos       %integer;   #IMPLIED
  featurerefs     IDREFS     #IMPLIED
  sites          CDATA      #IMPLIED
  on-strand       (plus|minus|source|auto-strand) #IMPLIED
  auto-plot       %yesorno;  #IMPLIED
  one-vert-col    %yesorno;  #IMPLIED
```

defcaption	CDATA	#IMPLIED
caption-font	IDREF	#IMPLIED
position-font	IDREF	#IMPLIED
align-text	(auto-align top center bottom)	#IMPLIED
text-length	%length;	#IMPLIED
wrap-text	%yesorno;	#IMPLIED
show-text	%yesorno;	#IMPLIED
show-position	%yesorno;	#IMPLIED
line-gap	%length;	#IMPLIED
text-gap	%length;	#IMPLIED
line-width	%length;	#IMPLIED
line-color	%color;	#IMPLIED
line-length	%length;	#IMPLIED>

Notes

<u>Attribute</u>	<u>Explanation</u>
featureref	id of feature
show-as-one	show as one site using defcaption and caption-font and group-pos
check-members	check display on for each
group-pos	if show-as-one=1
featurerefs	get sites from features or digest sets
sites	comma-separated site list
on-strand	locate on strand if true
auto-plot	plot automatically
one-vert-col	applies to vertical only
defcaption	default caption
caption-font	select by idref
position-font	font by idref
align-text	position to align text
text-length	maximum line length
wrap-text	wrap at max line
show-text	display caption
show-position	display position number
line-gap	default if not specified
text-gap	default if not specified
line-width	default if not specified
line-color	default if not specified
line-length	default if not specified

Interval-widget Element

Display interval on the sequence.

Content Model

```
<!ELEMENT Interval-widget (Object?, Resource*, %links;)>
```

Attributes


```

<!ATTLIST Interval-widget
  %attrs;
  featureref      IDREF          #IMPLIED
  startpos        %integer;      #REQUIRED
  endpos          %integer;      #REQUIRED
  show-start-err  %yesorno;      #IMPLIED
  start-err-len   %integer;      #IMPLIED
  show-end-err    %yesorno;      #IMPLIED
  end-err-len     %integer;      #IMPLIED
  on-sequence     CDATA          #IMPLIED
  gap-in-seq-line %yesorno;      #IMPLIED
  show-positions  (no|plus|minus) #IMPLIED
  start-pos-only  %yesorno;      #IMPLIED
  position-font   IDREF          #IMPLIED
  line-width      %linewidth;    #IMPLIED
  interior-gap    %length;       #IMPLIED
  fill-pattern    (clear|horiz|vert|fdiag|bdiag|
                  cross|diagcross|solid) #IMPLIED
  border-color    %color;        #IMPLIED
  fill-fg-color   %color;        #IMPLIED
  fill-bg-color   %color;        #IMPLIED
  auto-offset     %yesorno;      #IMPLIED
  offset-from-seq %length;       #IMPLIED
  arrow-start-len %length;       #IMPLIED
  arrow-end-len   %length;       #IMPLIED
  arrow-width     %length;       #IMPLIED
  bracket-text-gap %length;      #IMPLIED
  bracket-line-gap %length;      #IMPLIED
  bracket-line-len %length;      #IMPLIED
  bracket-line-wid %length;      #IMPLIED
  bracket-color   %color;        #IMPLIED
  plus-text       CDATA          #IMPLIED
  plus-font       IDREF          #IMPLIED
  plus-bracket    CDATA          #IMPLIED
  minus-text      CDATA          #IMPLIED
  minus-font      IDREF          #IMPLIED
  minus-bracket   CDATA          #IMPLIED
  center-text     CDATA          #IMPLIED
  center-font     IDREF          #IMPLIED
  start-type      CDATA          #IMPLIED
  end-type        CDATA          #IMPLIED>

```

Notes

<u>Attribute</u>	<u>Explanation</u>
featureref	id of feature
startpos	start of interval
endpos	end of interval
show-start-err	error bar at start
start-err-len	length of bar
show-end-err	error bar at end
end-err-len	length of bar
on-sequence	position on sequence = (center,plus,minus)
gap-in-seq-line	yes=gap around intvl
show-positions	show numbers

start-pos-only	show only start pos
position-font	font for numbers
line-width	border of rectangle
interior-gap	interior of rect
fill-pattern	fill of rect
border-color	color of rect border
fill-fg-color	interior foreground
fill-bg-color	interior background
auto-offset	find open area auto
offset-from-seq	distance from seq
arrow-start-len	arrow length if used
arrow-end-len	arrow length if used
arrow-width	width of either
bracket-text-gap	text to bracket
bracket-line-gap	bracket to rect
bracket-line-len	length of bracket
bracket-line-wid	width of bracket line
bracket-color	color of bracket line
plus-text	plus strand caption
plus-font	font for caption
plus-bracket	none,angled,tee (linear)
minus-text	minus strand caption
minus-font	font for caption
minus-bracket	none,angled,tee (linear)
center-text	caption in rectangle
center-font	font for caption
start-type	none,blunt,round,arrow
end-type	none,blunt,round,arrow

Blowup-widget Element

Display text, typically bases or residues, aligned to a particular place on the **View** line.

Content Model

```
<!ELEMENT Blowup-widget (Object?, Resource*, %links;)>
```

Attributes

```
<!ATTLIST Blowup-widget
  %attrs;
  featureref      IDREF      #IMPLIED
  startpos        %integer;   #REQUIRED
  endpos          %integer;   #REQUIRED
  on-plus         %yesorno;   #IMPLIED
  line-width      %linewidth; #IMPLIED
  interior-gap    %length;    #IMPLIED
  fill-pattern    (clear|horiz|vert|fdiag|bdiag|
                  cross|diagcross|solid) #IMPLIED
  border-color    %color;     #IMPLIED
```

fill-fg-color	%color;	#IMPLIED
fill-bg-color	%color;	#IMPLIED
bracket-text-gap	%length;	#IMPLIED
bracket-line-gap	%length;	#IMPLIED
bracket-line-len	%length;	#IMPLIED
bracket-line-wid	%length;	#IMPLIED
bracket-color	%color;	#IMPLIED
cut-thickness	%length;	#IMPLIED
plus-text	CDATA	#IMPLIED
plus-font	IDREF	#IMPLIED
minus-text	CDATA	#IMPLIED
minus-color	%color;	#IMPLIED
inside-text	CDATA	#IMPLIED
inside-font	IDREF	#IMPLIED
outside-text	CDATA	#IMPLIED
outside-font	IDREF	#IMPLIED>

Notes

<u>Attribute</u>	<u>Explanation</u>
featureref	id of feature
startpos	start of interval
endpos	end of interval
on-plus	position on sequence
line-width	see interval widget
plus-text	text inside blowup
plus-font	caption for text
minus-text	second line of text
minus-color	caption for text
inside-text	text opposite blowup
inside-font	font for text
outside-text	text outside blowup
outside-font	font for text

Aligned-chart-widget Element

Derives its data from a **Chart** element and displays a chart aligned to the view line.

Content Model

```
<!ELEMENT Aligned-chart-widget (Chart,Quantifier?,Object?,
Resource*,%links;)>
```

Attributes

```
<!ATTLIST Aligned-chart-widget
%attrs;
featureref          IDREF          #IMPLIED
full-only           %yesorno;      #IMPLIED
min-points           %integer;      #IMPLIED
field-number         %integer;      #IMPLIED
startpos             %integer;      #REQUIRED
endpos              %integer;      #REQUIRED
min-offset           %length;      #REQUIRED
```

max-offset	%length;	#REQUIRED
on-plus	%yesorno;	#IMPLIED
min-data	%real;	#REQUIRED
max-data	%real;	#REQUIRED
dec-places	%integer;	#IMPLIED
show-y-axis	%yesorno;	#IMPLIED
histogram	%yesorno;	#IMPLIED
line-width	%linewidth;	#IMPLIED
line-color	%color;	#IMPLIED
caption	CDATA	#IMPLIED
caption-font	IDREF	#IMPLIED
data-font	IDREF	#IMPLIED
standard-height	%yesorno;	#IMPLIED>

Notes

<u>Attribute</u>	<u>Explanation</u>
featureref	id of feature
full-only	show only all data
min-points	min points to show
field-number	if fields are space or comma-separated, indicate which field to read, else first
startpos	1st alignment pos on seq
endpos	last alignment pos on seq
min-offset	plot region start offset
max-offset	plot region end offset
on-plus	if yes, plot on plus side
min-data	lowest plotted data value
max-data	highest
dec-places	dec places in display
show-y-axis	yes=show y axis range
histogram	yes=plot points as histo
line-width	width of plotting line
line-color	color of plotting line
caption	chart caption
caption-font	font for caption
data-font	font for axis labels
standard-height	histogram + quantifier

View and View-related Elements

View Element

View of a **Sequence**.

Content Model

```
<!ELEMENT View (Numbering?,Margin?,%seqdisplaywidgets,,
                Object?, Resource*, %links;)>
```

Attributes

```

<!ATTLIST View
  %attrs;
  seqref      IDREF      #REQUIRED
  histref     IDREF      #IMPLIED
  title1      CDATA      #IMPLIED
  title1-font IDREF      #IMPLIED
  title2      CDATA      #IMPLIED
  title2-font IDREF      #IMPLIED
  startpos    %integer;  #IMPLIED
  endpos      %integer;  #IMPLIED>

```

Notes

<u>Attribute</u>	<u>Explanation</u>
seqref	sequence id
histref	histogram widget
title1	first title line
title1-font	font for first
title2	second title line
title2-font	font for second
startpos	first if not given
endpos	full if not given

Feature-histogram-widget Element

A histogram showing the distribution of features on a view.

Content Model

```

<!ELEMENT Feature-histogram-widget (Coord,Object?, Resource*, %links;)>

```

Attributes

```

<!ATTLIST Feature-histogram-widget
  %attrs;
  viewref      IDREF      #REQUIRED
  caption      CDATA      #IMPLIED
  capt-font    IDREF      #IMPLIED
  numb-font    IDREF      #IMPLIED
  features-histogram %integer; #IMPLIED
  feature-position %integer; #IMPLIED
  nbins        %integer; #IMPLIED
  height-per-bin %length;  #IMPLIED
  line-width   %linewidth; #IMPLIED
  bin-width    %length;  #IMPLIED
  line-color   %color;    #IMPLIED
  fill-bins    %yesorno;  #IMPLIED
  maximum-height %length;  #IMPLIED
  auto-fit     %yesorno;  #IMPLIED>

```

Notes

<u>Attribute</u>	<u>Explanation</u>
viewref	reference view
caption	caption for histo

capt-font	caption font
numb-font	number font
features-histogram	0 = none, 1 = all, 2 = hidden, 3 = showing
feature-position	0 = left end, 1 = right end, 2 = midpoint, 3 = range
nbins	number of bins
height-per-bin	height units
line-width	width of plotting
bin-width	per bin
line-color	plotting color
fill-bins	if true, solid
maximum-height	total allowed height
auto-fit	if true, set height of highest frequency to maximum-height

View-master-widget Element

A component for controlling simultaneous access to a set of views.

Content Model

```
<!ELEMENT View-master-widget (Coord,Object?, Resource*, %links;)>
```

Attributes

```
<!ATTLIST View-master-widget
  views          IDREFS          #IMPLIED
  width          %length;        #IMPLIED
  height         %length;        #IMPLIED
  outerwidth     %linewidth;     #IMPLIED
  outerborder    %color;         #IMPLIED
  outerfill      %color;         #IMPLIED
  innerwidth     %linewidth;     #IMPLIED
  innerborder    %color;         #IMPLIED
  innerfill      %color;         #IMPLIED
  shapeborder    %color;         #IMPLIED
  shapefill      %color;         #IMPLIED
  disabledborder %color;         #IMPLIED
  disabledfill   %color;         #IMPLIED
  shape          (circular|horizontal|vertical) #IMPLIED
  hcenter        %length;        #IMPLIED
  vcenter        %length;        #IMPLIED
  hoffset        %length;        #IMPLIED
  voffset        %length;        #IMPLIED
  numviews       %integer;       #IMPLIED
  linear-length  %length;        #IMPLIED
  circular-diam  %length;        #IMPLIED>
```

Notes

<u>Attribute</u>	<u>Explanation</u>
views	if none, all on page
width	overall width
height	overall height

outerwidth	dimensions and colors
hcenter	center - horiz
vcenter	center - vert
hoffset	to next view
voffset	to next view
numviews	# to show
linear-length	line length
circular-diam	circle diam

Sequence-viewer

A **Sequence-viewer** element controls the display in a separate viewer.

Class-key Element

Content Model

```
<!ELEMENT Class-key EMPTY>
```

Attributes

```
<!ATTLIST Class-key
  key-title      CDATA      #REQUIRED
  key-class      CDATA      #REQUIRED
  border-color   %color;    #IMPLIED
  fill-color     %color;    #IMPLIED
  row            %integer;   #IMPLIED
  enabled        %yesorno;   #IMPLIED
  set-number     %integer;   #IMPLIED>
```

Sequence-viewer Element

Content Model

```
<!ELEMENT Sequence-viewer (Class-key*,Object?, Resource*, %links;)>
```

Attributes

```
<!ATTLIST Sequence-viewer
  %attrs;
  refseq          IDREF      #IMPLIED
  maximize-window %yesorno;   #IMPLIED
  complete-seq    %yesorno;   #IMPLIED
  view-start      %integer;   #IMPLIED
  view-end        %integer;   #IMPLIED
  fit-to-window   %yesorno;   #IMPLIED
  spacing         %integer;   #IMPLIED
  bases-per-line  %integer;   #IMPLIED
  bases-per-window %integer;   #IMPLIED
  int-pos         %integer;   #IMPLIED
  interval-height-min %integer; #IMPLIED
```

interval-height-max	%integer;	#IMPLIED
font-pos	%integer;	#IMPLIED
font-size-min	%integer;	#IMPLIED
font-size-max	%integer;	#IMPLIED
show-sites	%yesorno;	#IMPLIED
sites-as-ints	%yesorno;	#IMPLIED
sites-not-full	%yesorno;	#IMPLIED
sites-by-pct	%yesorno;	#IMPLIED
sites-percent	%integer;	#IMPLIED
interval-pos	%integer;	#IMPLIED
interval-overlap	%yesorno;	#IMPLIED
interval-separate	%yesorno;	#IMPLIED
interval-by-width	%yesorno;	#IMPLIED
int-pixels	%integer;	#IMPLIED
int-as-point	%yesorno;	#IMPLIED
plus-strand	%yesorno;	#IMPLIED
minus-strand	%yesorno;	#IMPLIED
frame	%yesorno;	#IMPLIED
show-point-titles	%yesorno;	#IMPLIED
show-interval-titles	%yesorno;	#IMPLIED
interval-titles-above	%yesorno;	#IMPLIED
interval-title-once	%yesorno;	#IMPLIED
title-length	%integer;	#IMPLIED
show-key	%yesorno;	#IMPLIED
feature-histogram	%integer;	#IMPLIED
feature-position	%integer;	#IMPLIED
density-per-window	%yesorno;	#IMPLIED
density-method	%integer;	#IMPLIED
density-threshold	%integer;	#IMPLIED
nbins	%integer;	#IMPLIED
pixels-per-countX1000	%integer;	#IMPLIED
max-histo-height	%integer;	#IMPLIED
auto-fit-histogram	%yesorno;	#IMPLIED>

Notes

<u>Attribute</u>	<u>Explanation</u>
refseq	else get from user
maximize-window	max display window
complete-seq	show full seq
view-start	n/a if complete
view-end	n/a if complete
fit-to-window	whole view shown
spacing	0 to 10 pixels
bases-per-line	<=0 = use default
bases-per-window	<=0 = use default
int-pos	0 to 10 of min/max
interval-height- min	height in pixels
interval-height- max	height in pixels
font-pos	0 to 10 of min/max
font-size- min	size in points
font-size- max	size in points
show-sites	yes=show sites

sites-as-ints	sites as intervals
sites-not-full	if true, no sites when full view
sites-by-pct	plot sites only if view < % of full
sites-percent	percentages if true
interval-pos	0=none,1=online, 2=top,3=strand
interval-overlap	allow overlap
interval-separate	separate classes
interval-by-width	if yes, plot as interval if width
int-pixels	criterion or plot
int-as-point	if test fails, plot as point if true
	or not at all
plus-strand	plot features on
minus-strand	plot features on
frame	0=all, or 1,2,3
show-point-titles	only of int titles
show-interval-titles	plot text
interval-titles-above	if false, plot in box
interval-title-once	if true, show once
title-length	text length for plot
show-key	display class key
feature-histogram	none,all,hide,show
feature-position	start,end,mid,range
density-per-window	false=per 1000 pix
density-method	hide,plot,as points
density-threshold	cutoff
nbins	plotting bins
pixels-per-countX1000	pixels per countX1000
max-histo-height	maximum height,pixels
auto-fit-histogram	set highest bin max

Other Sequence-related Display Objects

Gel-widget Element

A **Gel-widget** element simulates a gel electrophoresis display using data from **Digest-set** elements.

Content Model

```
<!ELEMENT Gel-widget (Coord,Border?,Digest-set,Object?,
                        Resource*, %links;)>
```

Attributes

```
<!ATT LIST Gel-widget
  %attrs;
  digests          IDREFS          #REQUIRED
  title-font       IDREF           #IMPLIED
```

min-length	%integer;	#IMPLIED
max-length	%integer;	#IMPLIED
min-label	%integer;	#IMPLIED
scale-factor	%real;	#IMPLIED
label-decimals	%integer;	#IMPLIED
label-ratio	%real;	#IMPLIED
label-font	IDREF	#IMPLIED
gel-length	%length;	#IMPLIED
lane-width	%length;	#IMPLIED
lane-gap	%length;	#IMPLIED
lane-thick	%linewidth;	#IMPLIED
lane-color	%color;	#IMPLIED
band-prop	%yesorno;	#IMPLIED
band-thick	%linewidth;	#IMPLIED
band-low	%linewidth;	#IMPLIED
band-color	%color;	#IMPLIED
confusion	%linewidth;	#IMPLIED
confusion-color	%color;	#IMPLIED
plot-big	%yesorno;	#IMPLIED
use-key	%yesorno;	#IMPLIED
show-band-on-seq	%yesorno;	#IMPLIED>

Notes

<u>Attribute</u>	<u>Explanation</u>
digests	digests in sets
title-font	font for title
min-length	minimum frag length
max-length	maximum frag length
min-label	low axis position
scale-factor	scale label
label-decimals	for display
label-ratio	labelling ratio
label-font	font for axis
gel-length	length of gel
lane-width	width per lane
lane-gap	between lanes
lane-thick	lane border line; band thickness inversely proportional to mobility if true
lane-color	lane edge color
band-prop	proportional (thickness is used if not proportional or sets the maximum if band width is inversely proportional to fragment mobility)
band-thick	band thickness
band-low	thinnest band (band-low is width of the thinnest band if proportional)
band-color	color of bands
confusion	overlap test
	confusion = distance between bands used to label bands as confusable; if 0, bands must touch or overlap

confusion-color	color of confusable
plot-big	plot if off end
use-key	label bands in key
show-band-on-seq	if true, show temporary interval on sequence when band selected

Seq-data-widget Element

A **Seq-data-widget** element displays a range of sequence data.

Content Model

```
<!ELEMENT Seq-data-widget (Coord,Border?,Seq-data?,Object?,
                           Resource*, %links;)>
```

Attributes

```
<!ATTLIST Seq-data-widget
  %attrs;
  featureref      IDREF      #IMPLIED
  seqref          IDREF      #IMPLIED
  startpos        %integer;   #REQUIRED
  length          %integer;   #REQUIRED
  caption         CDATA      #IMPLIED
  seq-font        IDREF      #IMPLIED
  caption-font    IDREF      #IMPLIED
  bases-per-line  %integer;   #IMPLIED
  number-bases    %yesorno;   #IMPLIED
  space-blocks    %yesorno;   #IMPLIED
  double-stranded %yesorno;   #IMPLIED
  translate-dna   (yes|no|only) #IMPLIED
  translate-start  %integer;   #IMPLIED
  translate-length %integer;   #IMPLIED
  source          %url;       #IMPLIED
  seq-len         %integer;   #IMPLIED
  is-protein      %yesorno;   #IMPLIED>
```

Notes*

<u>Attribute</u>	<u>Explanation</u>
featureref	id of feature
seqref	id of sequence
startpos	offset on sequence
length	length of region
caption	caption for display
seq-font	font for data
caption-font	font for caption
bases-per-line	wrap after # bases
number-bases	yes=show numbers
space-blocks	yes=space every 10
double-stranded	yes=show 2 strands
translate-dna	yes/only=translate

translate-start	start of translate
translate-length	length of translate
source	source of seq data
seq-len	if url or Seq-data
is-protein	if url or Seq-data

* if there is a **Seq-data** element, get data there; if seqref is defined, get data from the sequence; otherwise use source file.

Sequence-Independent Display Widgets

Popup-text-widget Element

Content Model

```
<!ELEMENT Popup-text-widget (#PCDATA)>
```

Attributes

```
<!ATTLIST Popup-text-widget
  %attrs;>
```

Caption-widget Element

Display text on the page or in a separate window.

Content Model

```
<!ELEMENT Caption-widget (Coord, Resource*, %links;)>
```

Attributes

```
<!ATTLIST Caption-widget
  %attrs;
  text          CDATA          #IMPLIED
  font          IDREF          #IMPLIED
  orientation    (horizontal|down|angled) #IMPLIED
  angle          %integer;      #IMPLIED
  border        %yesorno;      #IMPLIED
  border-gutter  %length;       #IMPLIED
  border-width   %length;       #IMPLIED
  border-color   %color;        #IMPLIED
  wordwrap       %yesorno;      #IMPLIED
  wrap-length    %length;       #IMPLIED
  align          (left|center|right) #IMPLIED>
```

Notes

<u>Attribute</u>	<u>Explanation</u>
text	caption (textual content)
font	id of the font element
orientation	specifies manner in which text is to be drawn; default=horz

angle	specifies plotting angle, if orientation set to angled
border	specifies if line drawn around text; yes=border
border-gutter	specifies gutter around text if border="1"
border-width	specifies line width if border="1"
border-color	specifies color of border line if border="1"
wordwrap	yes=wrap
wrap-length	line length before wrapping if wordwrap="1"
align	if horizontal, apply alignment

Line-pointer-widget Element

Multi-segmented line with arrows; segments are defined by a list of at least two **Coord** elements.

Content Model

```
<!ELEMENT Line-pointer-widget (Coord+, Resource*, %links;)>
```

Attributes

```
<!ATTLIST Line-pointer-widget
  %attrs;
  line-style %linestyle;          #IMPLIED
  line-width %linewidth;         #IMPLIED
  line-color %color;             #IMPLIED
  use-start-arrow %yesorno;      #IMPLIED
  start-arrow-length %length;    #IMPLIED
  start-arrow-width %length;     #IMPLIED
  use-end-arrow %yesorno;        #IMPLIED
  end-arrow-length %length;      #IMPLIED
  end-arrow-width %length;       #IMPLIED
  caption CDATA                  #IMPLIED
  font IDREF                     #IMPLIED
  above-line %yesorno;           #IMPLIED>
```

Notes

<u>Attribute</u>	<u>Explanation</u>
line-style	solid, dash, dot
caption	if specified, the caption is displayed on first segment
above-line	if caption displayed

Shape-widget Element

Display a basic shape with or without a fill.

Content Model

```
<!ELEMENT Shape-widget (Coord, Resource*, %links;)>
```

Attributes

```
<!ATTLIST Shape-widget
  %attrs;
  height      %length;      #IMPLIED
  width       %length;      #IMPLIED
  filled      %yesorno;     #IMPLIED
  line-width  %linewidth;   #IMPLIED
  shape       %displayshape; #IMPLIED
  color       %color;       #IMPLIED>
```

Notes

<u>Attribute</u>	<u>Explanation</u>
height	overall height of shape
width	overall width of shape
filled	if yes, solid shape
line-width	if no, use line width
shape	see shape definition
color	color of shape

File-widget Element

Display the contents of a file that may contain graphic or text content.

Content Model

```
<!ELEMENT File-widget (Coord?,Border?,Image-map?,Object?,
  Resource*, %links;)>
```

Attributes

```
<!ATTLIST File-widget
  %attrs;
  own-window  %yesorno; #IMPLIED
  caption     CDATA     #IMPLIED
  height      %length;  #IMPLIED
  width       %length;  #IMPLIED
  text-font   IDREF     #IMPLIED
  type        CDATA     #IMPLIED
  source      %url;     #REQUIRED>
```

Notes

<u>Attribute</u>	<u>Explanation</u>
own-window	embed if false
caption	if own window
height	0=use graphic size
width	0=use graphic size
text-font	if text file
type	text,gif,jpeg,png,tif
source	file name or url

Table-column Element

Used by **Table-widget** to define data.

Content Model

```
<!ELEMENT Table-column EMPTY>
```

Attributes

```
<!ATTLIST Table-column
  colnum      %integer;          #REQUIRED
  colwid      %length;          #IMPLIED
  align       (left|center|right) "right">
```

Notes

<u>Attribute</u>	<u>Explanation</u>
colnum	data column
colwid	column width

Table-widget Element

Visualization of table data in their own window or on the page.

Content Model

```
<!ELEMENT Table-widget (Coord?,Border?,Table-column+,Object?,
                        Resource*, %links;)>
```

Attributes

```
<!ATTLIST Table-widget
  %attrs;
  own-window   % yesorno;      #IMPLIED
  own-in-grid  % yesorno;      #IMPLIED
  tableid      IDREF           #REQUIRED
  rownums      CDATA           #IMPLIED
  title-font   IDREF           #IMPLIED
  header-font  IDREF           #IMPLIED
  cell-font    IDREF           #IMPLIED
  row-height   %length;        #IMPLIED
  rules        %trules;        #IMPLIED
  rule-color   %color;         #IMPLIED
  rule-width   %linewidth;     #IMPLIED
  cellspacing  %length;        #IMPLIED
  cellpadding  %length;        #IMPLIED>
```

Notes

<u>Attribute</u>	<u>Explanation</u>
own-window	display in its own window
own-in-grid	if own window, grid or html
tableid	id of source table
rownums	optional comma-separated row numbers, else all rows

title-font	font for title
header-font	font for headers
cell-font	font for cells
row-height	else text height
rules	rulings between rows and cols
rule-color	color of rule lines
rule-width	width of rule lines
cellspacing	spacing between cells
cellpadding	spacing within cells

Text-key-item Element

Describes an item used in a **Text-key-widget**.

Content Model

```
<!ELEMENT Text-key-item EMPTY>
```

Attributes

```
<!ATTLIST Text-key-item
  key          CDATA #REQUIRED
  explanation  CDATA #REQUIRED>
```

Notes

<u>Attribute</u>	<u>Explanation</u>
key	term defined in key
explanation	definition of term

Text-key-widget Element

Explanatory key for text associated with an abbreviation.

Content Model

```
<!ELEMENT Text-key-widget (Coord,Border?,Text-key-item+,
                           Resource*, %links;)>
```

Attributes

```
<!ATTLIST Text-key-widget
  %attrs;
  key-font          IDREF      #IMPLIED
  explanation-font  IDREF      #IMPLIED
  key-width         %length;   #IMPLIED
  item-gap          %length;   #IMPLIED>
```

Notes

<u>Attribute</u>	<u>Explanation</u>
key-font	key text font
explanation-font	explanation text font

key-width	width of key text
item-gap	vertical gap

Fill-key-widget Element

Element used to display an explanatory key with text associated with filled rectangles.

Content Model

```
<!ELEMENT Fill-key-widget (Coord,Border?, Resource*, %links;)>
```

Attributes

```
<!ATTLIST Fill-key-widget
  %attrs;
  font                IDREF                #IMPLIED
  pattern-width       %length;              #IMPLIED
  pattern-height      %length;              #IMPLIED
  line-width          %linewidth;           #IMPLIED
  pattern-gap         %length;              #IMPLIED
  patterns            CDATA                 #IMPLIED
  fore-colors         CDATA                 #IMPLIED
  back-colors         CDATA                 #IMPLIED
  explanations        CDATA                 #IMPLIED>
```

Notes

<u>Attribute</u>	<u>Explanation</u>
font	text font
pattern-width	box width
pattern-height	box height
line-width	shading line width
pattern-gap	gap between boxes
patterns	comma-sep list of fills: clear,horiz,vert,fdiag,bdiag,cross,diagcross,solid
fore-colors	comma-sep list of colors
back-colors	comma-sep list of colors
explanations	comma-sep separated list

Symbol-key-widget Element

Displays a key that associates text with symbols.

Content Model

```
<!ELEMENT Symbol-key-widget (Coord,Border?, Resource*, %links;)>
```

Attributes

```
<!ATTLIST Symbol-key-widget
  %attrs;
  do-substitute %yesorno;  #IMPLIED
  font          IDREF      #IMPLIED
```

size	%length;	#IMPLIED
line-width	%linewidth;	#IMPLIED
color	%color;	#IMPLIED
open-circle	CDATA	#IMPLIED
closed-circle	CDATA	#IMPLIED
open-square	CDATA	#IMPLIED
closed-square	CDATA	#IMPLIED
triangle	CDATA	#IMPLIED
suppress	CDATA	#IMPLIED

Notes

<u>Attribute</u>	<u>Explanation</u>
do-substitute	no substitutions if false
font	font for symbol text
size	diameter of symbol
line-width	width if open
color	symbol color
open-circle	text to replace
closed-circle	
open-square	
closed-square	
triangle	
suppress	suppress if on list

Axis-widget Element

Free-standing axis that displays numerical scaling information on X-axis or Y-axis.

Content Model

```
<!ELEMENT Axis-widget (Coord,(X-axis|Y-axis))>
```

Attributes

```
<!ATTLIST Axis-widget
  %attrs;
  caption          CDATA          #IMPLIED
  capt-font        IDREF          #IMPLIED
  ascending        %yesorno;      #IMPLIED
  topright         %yesorno;      #IMPLIED
  axis-length      %length;       #REQUIRED
  start            %integer;       #REQUIRED
  end              %integer;       #REQUIRED
  num-divisions    %integer;       #IMPLIED
  show-first       %yesorno;      #IMPLIED
  zero-as-1       %yesorno;      #IMPLIED
  labels           CDATA          #IMPLIED>
```

Notes*

<u>Attribute</u>	<u>Explanation</u>
caption	caption for axis
capt-font	font for caption

ascending	numbering direc.
topright	above or right
axis-length	length of line
start	integer range
end	low to high
num-divisions	if not use-int
show-first	display number at start of line
zero-as-1	show position 0 as position 1
labels	label points

* if labels is defined, it is treated as a comma-separated list of labels to be shown at successive intervals, including the first; use ,, to indicate an unlabeled position

Chart Widget and Related Elements

X-chart-axis | Y-chart-axis Element

Elements used in chart display.

Content Model

```
<!ELEMENT (X-chart-axis|Y-chart-axis) EMPTY>
```

Attributes

```
<!ATTLIST (X-chart-axis|Y-chart-axis)
  %attrs;
  scale-automatic %yesorno;          #IMPLIED
  lower-limit     %real;              #IMPLIED
  upper-limit     %real;              #IMPLIED
  display-mode (decimal-places|best-fit|log|suppressed) #IMPLIED
  dec-places      %integer;           #IMPLIED
  grid (none|at-zero|at-ticks) #IMPLIED
```

Chart-page Element

Element used in chart display.

Content Model

```
<!ELEMENT Chart-page EMPTY>
```

Attributes

```
<!ATTLIST Chart-page
  %attrs;
  left %length; #IMPLIED
  top %length; #IMPLIED
  width %length; #IMPLIED
  length %length; #IMPLIED
  portrait %yesorno; #IMPLIED>
```

Chart-screen-display | Chart-print-display Element

Elements used in chart display.

Content Model

<!ELEMENT (Chart-screen-display|Chart-print-display) EMPTY>

Attributes

```
<!ATTLIST (Chart-screen-display|Chart-print-display)
  %attrs;
  monochrome           %yesorno;    #IMPLIED
  text-font-size       %integer;     #IMPLIED
  text-font-bold       %yesorno;     #IMPLIED
  numb-font-size       %integer;     #IMPLIED
  numb-font-bold       %yesorno;     #IMPLIED
  plot-thickness       %linewidth;   #IMPLIED
  frame-thickness      %linewidth;   #IMPLIED
  text-color           %color;       #IMPLIED
  numb-color           %color;       #IMPLIED
  plot-color           %color;       #IMPLIED
  frame-color          %color;       #IMPLIED
  plot-background-color %color;      #IMPLIED
  frame-background-color %color;     #IMPLIED
  symbol-size          %length;      #IMPLIED>
```

Chart-data Element

Element used to indicate the data source for a chart.

Content Model

<!ELEMENT Chart-data EMPTY>

Attributes

```
<!ATTLIST Chart-data
  %attrs;
  setnum               %integer;      #REQUIRED
  symbol (nosymbol|plus|ex|opencircle|
          filledcircle|opensquare|filledsquare) #IMPLIED
  line-format (none|line-only|symbol-only|both|err-bar) #IMPLIED
  line-color          %color;         #IMPLIED
  fill-color          color;          #IMPLIED
  do-fill             %yesorno;       #IMPLIED
  fill-pattern (clear|horiz|vert|fdiag|bdiag|
               cross|diagcross|solid) #IMPLIED
  legend              CDATA           #IMPLIED
  drop-left           CDATA           #IMPLIED
  drop-right          CDATA           #IMPLIED
  colrownum           %integer;       #IMPLIED
  values              DATA           #IMPLIED>
```

Notes

<u>Attribute</u>	<u>Explanation</u>
setnum	set 0 to 6 allowed
colrownum	number of row or col
values	if no table, comma-separated data

Chart-widget Element

A chart displayed on the page or in its own window.

Content Model

```
<!ELEMENT Chart-widget (Coord?,X-chart-axis?,
                        Y-chart-axis?,Chart-page?,Chart-screen-display?,
                        Chart-print-display?, Chart-data*,Object?,Resource*,%links;)>
```

Attributes

```
<!ATTLIST Chart-widget
  %attrs;
  own-window %yesorno; #IMPLIED
  width      %length;  #IMPLIED
  length     %length;  #IMPLIED
  chart-type (xy|multi-xy|box-plot|bar|histogram|line) #IMPLIED
  text-font  IDREF      #IMPLIED
  numb-font  IDREF      #IMPLIED
  title-1    CDATA      #IMPLIED
  title-2    CDATA      #IMPLIED
  x-title    CDATA      #IMPLIED
  y-title    CDATA      #IMPLIED
  legend-pos (below|upper-left|upper-right|
             lower-left|lower-right) #IMPLIED
  presort    %yesorno;  #IMPLIED
  x-no-zero  %yesorno;  #IMPLIED
  format     CDATA      #IMPLIED
  source     %url;      #IMPLIED
  tableid    IDREF      #IMPLIED
  bycolumn   %yesorno;  #IMPLIED
  column-starts CDATA    #IMPLIED
  column-widths CDATA    #IMPLIED
  header-count %integer; #IMPLIED>
```

Notes

<u>Attribute</u>	<u>Explanation</u>
x-no-zero	no zero position on x axis
--	if format not specified, data are supplied in chart-data --
format	acceptable = bsm html txt sdf db
source	none if in values
tableid	none if in values
bycolumn	none if in values
--	comma-separated lists of start positions and field widths --
column-starts	used for txt
column-widths	used for txt

header-count

txt skip lines

Page Element

Content Model

```
<!ELEMENT Page (Screen?,Paper?,Border?,  
(%pagedisplaywidgets;|%setwidgets;|View)*,Object?,Resource*, %links;) >
```

Attributes

```
<!ATTLIST Page  
  %attrs;>
```

Index of Elements

Aligned-chart-widget, 65, 160
Aligned-group, 28, 118
Aligned-groups, 28, 118
Aligned-pair, 28, 117
Aligned-sequence, 28, 117
Alignment-consensus, 29, 119
Alignment-point-set, 37, 112
Alignment-point-sets-widget, 69, 150
Alignment-summary, 28, 117
Analyses, 40, 128
Analysis, 40, 128
Arc, 37, 123
Arc-display, 70, 153
Area, 57, 142
Attribute, 5, 75
Attribute-list, 6, 76
Authority, 11, 84
Authority-domain, 11, 85
Authority-url, 11, 84
Axis-widget, 54, 175
Blowup-widget, 65, 159
Border, 47, 145
Bsml, 4, 89
Caption-widget, 51, 169
Case, 24, 105
Case-set, 25, 105
Cell-line, 15, 133
Chart, 65, 146
Chart-data, 60, 177
Chart-page, 60, 176
Chart-print-display, 60, 177
Chart-screen-display, 60, 177
Chart-widget, 60, 178
Children, 25, 107
Chromosome, 15, 91
Class-key, 66, 164
CLUSTAL-parameters, 28, 116
Contact-info, 8, 78
Content, 8, 82
Contributor, 6, 79
Coord, 46, 142
Coverage, 10, 81
Creator, 6, 79
Cross-reference, 13, 85
Cytoband, 15, 92
Date, 9, 82
Description, 41, 130
Development-stage, 42, 134
Digest-set, 20, 100
Display, 47, 137
Document-link, 14, 88
Dot-plot-widget, 69, 151
Enzyme, 41, 131
Equipment, 41, 131
Experiment, 41, 129
Experiments, 41, 129
Extended-link, 14, 87
Extrachromosomal-sequence, 15, 92
Feature, 18, 99
Feature-group, 21, 102
Feature-group-member, 21, 102
Feature-histogram-widget, 67, 162
Feature-table, 17, 102
Feature-tables, 17, 101
File-widget, 55, 171
Fill-key-widget, 59, 174
Font, 48, 138
Fonts, 48, 137
Format, 10, 80
Gel-widget, 61, 166
Genome, 15, 90
Genomes, 15, 90
Genotype, 25, 106
Genotype-set, 25, 105
Group-link, 14, 88
Haplotype, 25, 106
History, 10, 83
History-event, 10, 83
History-event-description, 11, 84
Identifier, 9, 83
Image-map, 57, 141
Info, 6, 76
Input-data, 40, 129
Interval-loc, 33, 98
Interval-widget, 64, 157
Isoform, 24, 104
Isoforms, 24, 103
Isoform-set, 25, 103

Language, 10, 81
Life-event, 25, 106
Line-pointer-widget, 53, 170
Link, 13, 86
Links, 48, 88
List-member, 26, 122
Locator, 14, 86
Log-entry, 39, 130
Margin, 46, 144
Materials, 41, 131
Mating, 25, 107
Measure, 41, 130
Measure-quantity, 41, 129
Measure-range, 41, 130
Measure-units, 41, 129
Member, 64, 147
Modification, 24, 97
Motif-data, 34, 109
Motif-element, 33, 111
Motif-table, 32, 112
Multiple-alignment-table, 28, 116
Network, 37, 122
Network-display-widget, 70, 152
Networks, 37, 122
Network-selection-set, 37, 123
Node, 37, 123
Node-display, 70, 152
Note, 41, 131
Numbering, 23, 33, 103
Object, 140
Organism, 15, 133
Organism-clone, 15, 133
Organization, 7, 78
Page, 43, 179
Pairwise-alignments, 28, 117
Paper, 46, 49, 145
Param, 141
Parameter, 40, 128
PCR-primer, 30, 119
PCR-primer-table, 30, 119
PCR-product, 30, 120
PCR-statistical-analysis, 30, 120
PCR-statistics, 30, 121
PCR-summary, 30, 120
Pedigree, 24, 107
Pedigree-set, 24, 107

Person, 7, 78
Phenotype, 24, 104
Phenotype-set, 104
Point-group-widget, 64, 156
Point-widget, 64, 155
Popup-text-widget, 50, 169
Postal-address, 8, 77
Primer-sequence, 30, 120
Product, 41, 132
Protocol, 42, 135
Protocol-step, 42, 136
Publisher, 9, 79
Qualifier, 18, 98
Quantifier, 45, 143
Queries, 40, 127
Query, 40, 127
Query-request, 40, 127
Query-return, 40, 128
Reagent, 41, 132
RefAuthors, 17, 100
Reference, 17, 99
RefJournal, 17, 100
RefTitle, 17, 100
Relation, 10, 80
Research, 39, 125
Resource, 6, 77
Restriction-enzyme, 20, 101
Rights, 10, 81
Sample, 41, 132
Screen, 46, 49, 144
Search, 39, 125
Search-conditions, 39, 125
Searches, 39, 125
Search-result-field, 39, 127
Search-result-field-desc, 39, 126
Search-result-item, 39, 126
Search-result-map, 39, 126
Search-result-set, 39, 126
Segment, 24, 93
Segment-set, 24, 93
Seq-data, 21, 96
Seq-data-import, 21, 96
Seq-data-widget, 62, 168
Seq-pair-alignment, 36, 113
Seq-pair-run, 36, 114
Sequence, 16, 94

Sequence-alignment, 29, 118
Sequence-data, 119
Sequence-import, 22, 95
Sequence-motif, 34, 110
Sequences, 16, 93
Sequence-search-output, 28, 116
Sequence-search-table, 28, 115
Sequence-viewer, 66, 164
Set, 26, 121
Sets, 26, 121
Shape-widget, 54, 170
Simple-set-widget, 67, 147
Site-loc, 18, 97
Software-system, 7, 79
Solution, 42, 134
Source, 10, 81
Strain, 15, 133
Structure, 42, 135
Style, 48, 139
Style-import, 48, 139
Styles, 48, 139
Subject, 10, 80
Supplier, 42, 135
Symbol, 47, 145
Symbol-key-widget, 59, 174
Table-column, 58, 172
Table-data, 27, 109
Table-import, 27, 108
Tables, 26, 108
Table-widget, 57, 172
Terminology, 42, 135
Text-key-item, 173
Text-key-widget, 58, 173
Tissue-sample, 42, 134
Title, 9, 80
Total-alignment, 118
Tree-node, 68, 148
Tree-set-widget, 68, 149
Type, 9, 80
Version, 9, 82
View, 49, 63, 161
View-axis-widget, 63, 155
View-line-widget, 63, 153
View-master-widget, 66, 163
X-axis, 47, 146
X-chart-axis, 60, 176
Y-axis, 47, 146
Y-chart-axis, 60, 176