APPENDIX 1: VAXML SPECIFICATION

VAXML (Virtual Anatomy eXtensible Markup Language) is a simple markup-language designed to ease the file-format problems associated with the sharing of 'virtual specimen' data. VAXML datasets use one or more STL/PLY files (either binary or ascii) to define the geometry of objects that comprise the dataset, together with one VAXML file that provides data on the dataset as a whole, and specifies how the STL files are to be put together.

An XML Schema file is given below; this provides a machine-readable definition of the format of XML files, and can be used to check the validity of any hand-written VAXML file. An English definition follows subsequently; this assumes a passing familiarity with XML syntax. These definitions are for VAXML version 2.

VAXML XML SCHEMA

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="vaxml">
    <xs:complexType>
      <xs:sequence>
        <xs:element maxOccurs="1" minOccurs="1" name="header">
          <xs:complexType>
            <xs:sequence>
              <xs:element maxOccurs="1" minOccurs="1" name="version" type="xs:integer"/>
              <xs:element maxOccurs="1" minOccurs="1" name="title" type="xs:string"/>
              <xs:element maxOccurs="1" minOccurs="0" name="scale" type="xs:float"/>
              <xs:element minOccurs="0" maxOccurs="unbounded" name="comments" type="xs:string"/>
              <xs:element minOccurs="0" maxOccurs="unbounded" name="reference" type="xs:string"/>
              <xs:element minOccurs="0" maxOccurs="unbounded" name="author" type="xs:string"/>
              <xs:element minOccurs="0" maxOccurs="unbounded" name="provenance" type="xs:string"/>
              <xs:element minOccurs="0" maxOccurs="unbounded" name="specimen" type="xs:string"/>
              <xs:element minOccurs="0" maxOccurs="unbounded" name="classification">
                <xs:complexType>
                  <xs:all>
                    <xs:element maxOccurs="1" minOccurs="1" name="rank" type="xs:string"/>
                    <xs:element maxOccurs="1" minOccurs="1" name="name" type="xs:string"/>
                  </xs:all>
                </xs:complexType>
              </xs:element>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element maxOccurs="1" minOccurs="0" name="groups">
          <xs:complexType>
            <xs:sequence>
```

```
      <xs:element minOccurs="0" maxOccurs="unbounded" name="group">
        <xs:complexType>
          <xs:all>
            <xs:element maxOccurs="1" minOccurs="1" name="name" type="xs:string"/>
            <xs:element name="key" maxOccurs="1" minOccurs="0">
              <xs:simpleType>
                <xs:restriction base="xs:string">
                  <xs:pattern value="[0-9A-Za-z]"/>
                </xs:restriction>
              </xs:simpleType>
            </xs:element>
            <xs:element maxOccurs="1" minOccurs="0" name="position" type="xs:integer"/>
            <xs:element maxOccurs="1" minOccurs="0" name="visible" type="xs:boolean"/>
            <xs:element maxOccurs="1" minOccurs="0" name="ingroup" type="xs:string"/>
          </xs:all>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element maxOccurs="1" minOccurs="1" name="objects">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="1" maxOccurs="unbounded" name="object">
        <xs:complexType>
          <xs:all>
            <xs:element maxOccurs="1" minOccurs="1" name="name" type="xs:string"/>
            <xs:element name="key" maxOccurs="1" minOccurs="0">
              <xs:simpleType>
                <xs:restriction base="xs:string">
                  <xs:pattern value="[0-9A-Za-z]"/>
                </xs:restriction>
              </xs:simpleType>
            </xs:element>
            <xs:element maxOccurs="1" minOccurs="0" name="position" type="xs:integer"/>
            <xs:element maxOccurs="1" minOccurs="0" name="visible" type="xs:boolean"/>
            <xs:element maxOccurs="1" minOccurs="0" name="ingroup" type="xs:string"/>
            <xs:element maxOccurs="1" minOccurs="1" name="file" type="xs:string"/>
            <xs:element maxOccurs="1" minOccurs="0" name="url" type="xs:string"/>
            <xs:element maxOccurs="1" minOccurs="0" name="matrix">
              <xs:complexType>
                <xs:sequence>
                  <xs:element maxOccurs="1" minOccurs="1" name="m0" type="xs:float"/>
                  <xs:element maxOccurs="1" minOccurs="1" name="m1" type="xs:float"/>
                  <xs:element maxOccurs="1" minOccurs="1" name="m2" type="xs:float"/>
                  <xs:element maxOccurs="1" minOccurs="1" name="m3" type="xs:float"/>
                  <xs:element maxOccurs="1" minOccurs="1" name="m4" type="xs:float"/>
                  <xs:element maxOccurs="1" minOccurs="1" name="m5" type="xs:float"/>
                  <xs:element maxOccurs="1" minOccurs="1" name="m6" type="xs:float"/>
                  <xs:element maxOccurs="1" minOccurs="1" name="m7" type="xs:float"/>
                  <xs:element maxOccurs="1" minOccurs="1" name="m8" type="xs:float"/>
                  <xs:element maxOccurs="1" minOccurs="1" name="m9" type="xs:float"/>
                  <xs:element maxOccurs="1" minOccurs="1" name="m10" type="xs:float"/>
```

```
            <xs:element maxOccurs="1" minOccurs="1" name="m11" type="xs:float"/>
            <xs:element maxOccurs="1" minOccurs="1" name="m12" type="xs:float"/>
            <xs:element maxOccurs="1" minOccurs="1" name="m13" type="xs:float"/>
            <xs:element maxOccurs="1" minOccurs="1" name="m14" type="xs:float"/>
            <xs:element maxOccurs="1" minOccurs="1" name="m15" type="xs:float"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element maxOccurs="1" minOccurs="1" name="material">
        <xs:complexType>
          <xs:all>
            <xs:element name="transparency" maxOccurs="1" minOccurs="0">
              <xs:simpleType>
                <xs:restriction base="xs:float">
                  <xs:minInclusive value="0"/>
                  <xs:maxInclusive value="1"/>
                </xs:restriction>
              </xs:simpleType>
            </xs:element>
            <xs:element maxOccurs="1" minOccurs="1" name="colour">
              <xs:complexType>
                <xs:all>
                  <xs:element name="red" maxOccurs="1" minOccurs="1">
                    <xs:simpleType>
                      <xs:restriction base="xs:integer">
                        <xs:minInclusive value="0"/>
                        <xs:maxInclusive value="255"/>
                      </xs:restriction>
                    </xs:simpleType>
                  </xs:element>
                  <xs:element name="green" maxOccurs="1" minOccurs="1">
                    <xs:simpleType>
                      <xs:restriction base="xs:integer">
                        <xs:minInclusive value="0"/>
                        <xs:maxInclusive value="255"/>
                      </xs:restriction>
                    </xs:simpleType>
                  </xs:element>
                  <xs:element name="blue" maxOccurs="1" minOccurs="1">
                    <xs:simpleType>
                      <xs:restriction base="xs:integer">
                        <xs:minInclusive value="0"/>
                        <xs:maxInclusive value="255"/>
                      </xs:restriction>
                    </xs:simpleType>
                  </xs:element>
                </xs:all>
              </xs:complexType>
            </xs:element>
          </xs:all>
        </xs:complexType>
      </xs:element>
    </xs:all>
```

```
              </xs:complexType>
            </xs:element>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
 </xs:element>
</xs:schema>
```

HUMAN-READABLE VAXML SPECIFICATION

Overall structure

All VAXML files begin with the line

<?xml version='1.0'?>

And subsequently contain a single <*vaxml*> element. This <*vaxml*> element contains (in sequence) a <*header*> element (required), a <*groups*> element (optional), and an <*objects*> element (required).

<header> element

The <*header*> element contains the following elements; these must occur in the order given below, but some are optional ([*optional*]), and others can occur more than once ([*repeat*])

<*version*>	An integer specifying the VAXML version used; currently this should be 2.
<*title*>	Text giving a title for the VAXML dataset
<*scale*> [*optional*]	A decimal value specifying how many millimetres there are in one STL/PLY unit (after transformation of objects by their specified matrices, see below). If omitted a value of '1' will be assumed by SPIERSview; other software might treat the scale as 'undefined' in these cases.
<*comments*> [*repeat*]	Text defining comments attached to the VAXML dataset; comments are textual notes that do not fit into any of the categories below
<*reference*> [*repeat*]	Text defining a reference attached to the VAXML dataset; references should be either to published or online work describing or in some way relevent to the specimen or taxon described by the dataset
<*author*> [*repeat*]	Text defining an author of the VAXML dataset
<*provenance*> [*repeat*]	Text defining an item of provenance information attached to the VAXML dataset; such items might include geographical information, repository information, and/or geological provenance and age.

<*specimen*> [*repeat*] Text defining an item of information concerning the specimen described by the VAXML dataset; such items might include accession numbers and details of preparation methodology.

<*classification*> [*repeat*] Text defining a taxon to which the VAXML dataset is referred. Classification items comprise two (required) elements, a <*rank*> (e.g. 'Genus') and a <*name*> (e.g. 'Homo')

<groups> element

The <*groups*> element is optional; it can legally be entirely omitted if the dataset does not make use of groups. If present, it comprises one or more <*group*> elements, each of which contains the following elements; these must occur in the order given below, but some are optional ([*optional*]).

<*name*> Text providing a name for the group; names must be unique within a VAXML file.

<*key*> [*optional*] A single character 0-9, A-Z or a-z specifying a keyboard shortcut key for the group. SPIERSview uses these keys to toggle visibility; other software might treat them differently or ignore them.

<*position*> [*optional*] An integer used to determine ordering of groups and objects in VAXML viewing software; this refers to ordering in lists (e.g. the SPIERSview Object Panel) rather than any sort of graphical ordering. Within any particular group, objects and groups should appear in ascending order of their <*position*>.

<*visible*> [*optional*] Either '0' (specifying a hidden group) or '1' (specifying a visible group).

<*ingroup*> [*optional*] Text specifying the <*name*> of the group in which this group belongs; if omitted the group belongs at root level (not in a group).

<objects> element

The <*objects*> element comprises one or more <*object*> elements, each of which contains the following elements; these must occur in the order given below, but some are optional ([*optional*]).

<*name*> Text providing a name for the object.

<*key*> [*optional*] A single character 0-9, A-Z or a-z specifying a keyboard shortcut key for the object. SPIERSview uses these keys to toggle visibility; other software might treat them differently or ignore them.

<*position*> [*optional*] An integer used to determine ordering of groups and objects in VAXML viewing software; this refers to ordering in lists (e.g. the SPIERSview Object Panel) rather than any sort of graphical ordering. Within any particular group, objects and groups should appear in ascending order of their <*position*>.

<*visible*> [*optional*] Either '0' (specifying a hidden object) or '1' (specifying a visible object).

<*ingroup*> [*optional*] Text specifying the <*name*> of the group in which this object belongs; if omitted the object belongs at root level (not in a group).

<*file*> Text providing the filename of the STL or PLY file specifying the geometry of this object. The filename may include a relative path; '/' should be used as a directory-separator rather than '\'.

<*url*> [*optional*] Text providing a URL ('web address') from which the STL or PLY file can be downloaded if not available on the local system. Viewing software opening the VAXML file may retrieve files from this location if they are not already present on the local system; SPIERSview provides such a facility. This provides a mechanism for VAXML delivery where only a single file need be downloaded through a web-browser.

<*matrix*> [*optional*] The <*matrix*> element defines a 4x4 matrix that specifies any three-dimensional transformations that should be made to the STL file before viewing. The element consists of 16 sub-elements, <m0> through <m15>, which specify the 16 decimal values comprising the matrix.

<*material*> The <*material*> element specifies details of the virtual material to be used to render this object. For PLY objects with vertex colour information the material is ignored by VAXML renderers. This data is encapsulated within this element to simplify expansion in future versions of VAXML; elements within the <*material*> element can be in any order, and unknown elements should simply be ignored by viewing software. Elements within the <*material*> element currently are:

- <*transparency*> [*optional*] Decimal number between 0 and 1 specifying degree of translucency (0.99 is very translucent, 0.01 hardly at all).
- <*colour*> Specifies the colour of the material; Americans should note the English spelling! The <*colour*> element comprises the three required sub-elements <*red*>, <*green*> and <*blue*>, each of which should contain an integer in the range 0-255.