

VOLCANOGASML: A FORMAT TO EXCHANGE GEOCHEMICAL VOLCANIC GASES DATA

Eric Reiter^{1*}

¹2, Rue des Abeilles, F-57440 Algrange, France
Email: ereiter@free.fr

ABSTRACT

Chemical analyses of volcanic gases consist of: location of sampling, date of sampling, identification of the sampling, etc. Nowadays, these data are generally represented in different formats. All of these formats are inflexible and machine dependent. XML has become the most important method of transferring data between computers. VolcanoGasML is a new format, based on XML, for the chemical analyses of volcanic gases. Its definition is divided into several layers: the first one describes the general information concerning the sample, the second, which is organized in several sublayers, contains the chemical data.

Keywords: Volcanic gases, XML, XSL, Data exchange, Geochemistry

1. INTRODUCTION

Chemical analyses of volcanic gases consist of: location of sampling, date of sampling, identification of the sampling, etc. Nowadays, these data are generally represented either in a binary format or in an Excel sheet. Both of these formats are inflexible and are machine dependent. In addition they are not designed for standardized reading. While in the past computer speed and storage capacities were a strong argument in favor of binary representation of data, they are no longer a limiting factor today.

Since its introduction in 1996, eXtensible Markup Language (XML) has become the single most important method of transferring data between computers. XML essentially allows developers to write their own schemas describing their data, which can then be parsed into a wide variety of viewers. Because of its extensible definition capabilities, its wide acceptance, and the existing large number of utilities and libraries for XML, a structured representation of all sorts of geochemical gas data should, in our opinion, be developed by defining a 'VolcanoGasML' standard. Such standard already exists for other geoscience domains (Schorlemmer, Wyss, Maraini, Weimer, & Baer, 2004).

Such a 'VolcanoGasML' standard, properly defined as a multi-layer definition, will provide the community with one single standard format covering location, date and composition data, according to the needs of the user. A 2-layer definition of VolcanoGasML is proposed: Layer 1 provides parameter data such as the volcano's name and identification of the sample. Additional information can optionally be added. Layer 2 contains chemical data. This layer is organized in several sub-layers, such as for general information (temperature, pH), chemical content, isotope data, and some ratios. All these data are optional and permit the user to select only those that are needed. The chemical content layer is divided into two additional sub-layers: one for the major element and one for the trace species.

2. VOLCANIC GASES

Magma contains dissolved gases that are released into the atmosphere during eruptions. Gases are also released from magma that either remains below ground (for example, as an intrusion) or is rising toward the surface. In such cases, gases may escape continuously into the atmosphere from the soil, volcanic vents, fumaroles, and hydrothermal systems.

2.1 Types of volcanic gases

The most abundant gases typically released into the atmosphere from volcanic systems are water vapor (H₂O), followed by carbon dioxide (CO₂) and sulfur dioxide (SO₂) (Bardindzeff, 1992). Volcanoes also release smaller amounts of other gases, including hydrogen sulfide (H₂S), hydrogen (H₂), carbon monoxide (CO), hydrogen chloride (HCl), hydrogen fluoride (HF), and helium (He), as defined in Table 1 for three well-known volcanoes. According to Delemelle and Stix (2000), additional gases have been identified, including HBr, Rn, Xe, He, Kr, Ar.

	Kilauea Summit	Erta` Ale	Momotombo
H ₂ O	37.1	77.2	97.1
CO ₂	48.9	11.3	1.44
SO ₂	11.8	8.34	0.50
H ₂	0.49	1.39	0.70
CO	1.51	0.44	0.01
H ₂ S	0.04	0.68	0.23
HCl	0.08	0.42	2.89
HF	---	---	0.26

Table 1: Examples of volcanic gas compositions, in volume percent concentrations (Symonds, Rose, Bluth, & Gerlach, 1994)

2.2 Why study volcanic gas?

As Japanese geochemist, Sadao Matsuo, has said, "Volcanic gas is a telegram from the earth's interior." Volcanic gas studies may be used to determine volatile and magma compositions, as well as the mineralogy of the underlying area. Similarly, different stages of volcanic activity can be deciphered by analyzing volcanic gas. An objective in gas monitoring is to determine changes in the release of certain gases from a volcano, chiefly carbon dioxide and sulfur dioxide. Such changes can be used with other monitoring information to provide eruption warnings and to improve our understanding of how volcanoes work.

Another area of specialty involves the study of volcanic sulfur gas emissions. The understanding of natural sulfur emissions applies to research of anthropogenic sulfur gases, which are known to contribute to acid rain and other pollution problems.

3. WHAT IS XML?

XML is a markup language for documents. A markup language is a mechanism to identify structures in a document. The XML specification defines a standard way to add markup to documents (Bray et al., 2006). The number of applications currently being developed that are based on, or make use of, XML documents is truly amazing (a popular example of XML use is Google Earth¹).

4. WHY XML?

One of the major problems related to data exchange in volcanology arises from the different needs in storing information. Although many geochemical data are common in most volcanological studies, these studies differ in their range of stored data, making a common format almost impossible until now. In order to achieve one single format for data interchange, the underlying technique must allow for user specific

¹ Google Earth is a virtual map of the world which shows the images, recorded by satellite, of most of the places of the Earth. That the visited place is perceived since the space or in some metres of the ground, the precision of photos is surprising. Google Earth can be downloaded at <http://earth.google.com/download-earth.html>

extensions without compromising the format definition or without making the data files unreadable for other users. This restriction prohibits the use of tabulator separated or column-oriented ASCII-files.

A future, more versatile, format should meet additional requirements. The use of open standards is proposed in order to make the implementation platform and system independent. Furthermore, open source software and multi-platform tools are available for working with data in the new format. This is important in order to ensure royalty-free access to the software that is needed to perform scientific work with the new format.

XML was selected for the format definition because it meets all our requirements and is already widely used in scientific applications. The switch to an XML data representation offers several advantages. The volcanological community is traditionally quick to reconsider the computational setup, procedures and data handling as new technologies emerge. With the omnipresence of the Internet, data exchange has become a natural and easy procedure. However, remarkably the geosciences community still use the old data formats and data exchange procedures. In recent years, the World Wide Web Consortium (W3C) has developed numerous standards and recommendations for data representation and handling. They reflect the increasingly recognized needs for easy and flexible data exchange. Basically, XML is the center point of these technologies. It is not only a meta-language to describe object-oriented data representation designed for the use in the Internet. It is more; XML is probably the most flexible data representation. Its main advantages are:

- **Tagged ASCII-files:** The information is coded with tags. This makes XML files human (and machine) readable and platform independent.
- **XML Schema (XSD):** Schemas, themselves expressed in XML, provide a comprehensive format definition language for describing own XML formats. They can be used to validate XML files with a parser.
- **Parser:** A parser is a program that analyses the grammatical structure of an input, with respect to a given formal grammar, in this case the schema. Open source parsing and validating tools are available for many platforms as well as for many programming languages. Most XML parsers use the platform- and language-neutral interfaces Simple API for XML (SAX) or the Document Object Model (DOM) (Wood et al., 1999) to parse an XML document into objects of a programming language. A great variety of such interfaces exists for most programming languages, offering a professional toolkit for working with XML files.
- **Individual extensibility:** Any XML-definition can readily integrate additional data. This makes individual extensions of VolcanoGasML possible without compromising the validity. Considering the aforementioned layers as extensions, any program dealing with a certain layer of our definition can use any catalog with higher layer definitions. For example, import routines supporting layer 1 can import layer 2 files without any modification while ignoring the additional data fields.
- **Stylesheet transformation (XSLT):** With XSLT, any XML-file can be transformed into another XML file (e.g., separating certain values, performing queries), into HTML pages for websites or web applications, or into simple ASCII files (CSV style) for importing data into existing programs. XSLT uses eXtended Stylesheet Language (XSL) files for instruction. Using XSL-FO (formatting objects), PDF or RTF output is possible. No complex programs have to be written for transforming the information into web-suitable formats.
- **Binding:** Binding provides a fast and convenient way to bind XML Schemas to a programming language object model, making it easy for developers to incorporate XML data and processing functions into applications. The binding API translates the XML schema definitions into an object model of a programming language. Several binding APIs are available: e.g., JAXB (Java), the Castor Project (Java), and LMX (C++).

In general, XML data files can be used to store data. When dealing with relatively large amounts of data, as commonly done in volcanological observatories, simple file handling becomes unsuitable. In this case, the

use of XML databases should be considered. Even SQL databases can be used for data storage, either by developing suitable import and export filters, or by using an XML-wrapper component that converts XML files into relational data structures. In the latter case, the SQL database behaves as an XML database. Many database applications provide XML support nowadays. Considering the fact that most observatories that use databases are already storing their data in SQL databases, import and export filters seem to be the appropriate solution. VolcanoGasML itself is mainly designed for information exchange not as a storage format.

5. ANATOMY OF AN XML FILE

XML documents are composed of markup and content. There are six kinds of markup that can occur in an XML document: elements, entity references, comments, processing instructions, marked sections, and document type declarations (Figure 1).

```
<?xml version="1.0"?>
<VolcanoGasML>
  <location>
    <volcano_name>Vulcano</ volcano_name >
  </location>
  <chemical>
    <T>211.2 </T>
  </chemical>
  <pH/>
</VolcanoGasML>
```

Figure 1: Anatomy of a XML file

The document begins with a processing instruction: `<?xml ...?>`. This is the *XML declaration*. While it is not required, its presence explicitly identifies the document as an XML document and indicates the version of XML according to which it was written.

Empty elements (`<pH/>` in this example) have a modified syntax. While most elements in a document are wrappers around some content, empty elements are simply markers where something occurs. The trailing `/>` in the modified syntax indicates to a program processing the XML document that the element is empty and no matching end-tag should be sought. Because XML documents do not require a document type declaration, without this clue it could be impossible for an XML parser to determine which tags were intentionally empty and which had been left empty by mistake.

XML has softened the distinction between elements that are declared as `EMPTY` and elements that merely have no content. In XML, it is legal to use the empty-element tag syntax in either case. It is also legal to use a start-tag/end-tag pair for empty elements: `<pH></pH>`. If interoperability is of any concern, it is best to reserve and use empty-element tag syntax for elements that are declared as `EMPTY`.

XML documents are composed of markup and content. The following sections introduce each of the six markup concepts in terms of VolcanoGasML. The detailed VolcanoGasML is fully given in Appendix 1. An example using this schema is given in Appendix 2.

6. VOLCANOGASML

The VolcanoGasML definition, described in the XSD schema language, is divided into several layers:

- Layer 1 describes the general information concerning the sample: the volcano name, the country where the volcano is located and/or geographical coordinates of the volcano (necessary when two

volcanoes around the world have the same name), the identification number of the sample, the date when the sample was taken, a location for the sample (e.g. northern part of the crater rim), a GPS location, and the sampling method. Two pieces of information are required: the volcano name and the identification number. The rest is optional.

- Layer 2 describes the chemical data, also organized in several sub-layers:
 - General information such as temperature and pH.
 - Chemical content which is subdivided into two other sub-layers:
 - Major species according to quantity such as H₂O, CO₂, HCl, SO₂, etc.
 - Trace species for the minor elements such as He, Ar, NH₃, PH₃ (Obenholzner et al., 2006).
 - Isotopic analyses
 - Ratios for some calculated ratios, e.g. Gas/Steam
- All the information for this layer is optional.

7. ADVANTAGE OF USING XML AND VOLCANOGASML

Below some of the advantages of XML and VolcanoGasML are highlighted, respectively:

7.1 Historic catalogues

Missing data in catalogs of historic geochemistry causes problems when using fixed-column formats. In VolcanoGasML, any information except the volcano's name and the sample's identification is optional and can be extended by error information to any length and precision.

7.2 XSLT

The eXtensible Stylesheet Transformation (XSLT) supports the concept of separating data from its presentation. While the data is stored in XML files, these files are not meant for presenting the data. This task can be accomplished in a very convenient way using stylesheets (XSL) and XSLT. The main advantage in this approach is the availability of fully developed XSLT engines. Only the stylesheets (an XML file again) need to be designed. With these stylesheets, the XSLT processor can generate almost any desired target format (a new XML file, HTML to publish it on the web or an intranet, ASCII to import the data into existing software, SVG to create graphs, RSS for news feeds or PDF)

XSLT-processors are readily available for many platforms (some under open-source licenses) and do not require development on the user's side. With XSLT, any geochemical information in VolcanoGasML can automatically be transformed into multiple formats, for web presentation, news feeds, bulletins etc.

7.3 Individual extensions of VolcanoGasML

Any XML definition can be extended in two ways, either by including additional user specific fields (tags) or by including itself into another XML definition. VolcanoGasML files can be included in any given XML file or can reference pictures which could be easily inserted in a target format.

8. VOLCANOGASML FITS: EXTENSION AND CUSTOMIZATION

As mentioned before, our basic VolcanoGasML definition can be extended with additional data fields (tags) to customize it according to different needs. An example of such a customization is discussed below.

Consider how to add electron microscopy data (images and chemical spectra) to a certain fumarole vent?

The extension consists of adding a tag in the VolcanoGasML Schema such as:

```
<xd:element name="picture" minOccurs="0" maxOccurs="unbounded">
```

```
<xd:complexType mixed="true">
  <xd:attribute name="path" use="required"/>
</xd:complexType>
</xd:element>
```

The path attribute is needed to tell the program where the picture is (on a computer or on the internet).

Then, the VolcanoGasML instance is as following:

```
<VolcanoGasML>
  <sample>
    <location>
      <volcano_name>Vulcano</volcano_name>
      <country>Southern Italy</country>
      <sample_date>1923</sample_date>
    </location>
    <chemical>
      <T>480</T>
      <major_elements>
        <h2o unit="% vol.">96.5</h2o>
        <so2 unit="% vol.">21.2</so2>
        <co2 unit="% vol.">55.2</co2>
        <hcl unit="% vol.">1.2</hcl>
      </major_elements>
    </chemical>
    <picture path="C:\mypicture.jpg"/>
  </sample>
</VolcanoGasML>
```

9. REMARKS

The prepared XML shown for Volcanic Gases is able to integrate other natural data that can be attributed to a coordinate, such as water analyses and petrological data. Unfortunately, this type of extension complicates maintenance of the schema and of the resulting data files. Using other XML formats for additional natural data is the best way to create specific schemas.

10. OUTLOOK

Almost all of the currently used format files (such as Excel) can be converted easily into VolcanoGasML files. However, additional work is needed. First, VolcanoGasML needs a user-friendly interface to create the XML file. This can be an HTML application. Second, some XSL/XSLT style sheets have to be developed to present the results in different formats. The first supported one should be PDF. The other common ones should (HTML, SVG) come later. Style sheets for RSS and ASCII formats will be developed only if the users ask for them. Third, tools which transform the currently used format into VolcanoGasML files have to be developed.

11. ACKNOWLEDGEMENTS

A special thanks to J. H. Obenholzner, Museum of Natural History of Vienna, Austria, who provided scientific literature concerning volcanic gases, stimulated this project and corrected this paper.

12. REFERENCES

- Bardintzeff J.-M., (1998) *Volcanologie*. 2e édition, Paris, France
- Bird, L., Goodchild, A., and Halpin, T., Object role modelling and XML-Schema, *Tech. rep.*, The University of Queensland, Australia, 2000.
- Bray T., Sperberg-Mc Queen C.M., Maler F., Yergeau F., Cowan J., Extensible Markup Language (XML) 1.0 (Fourth Edition) W3C Recommendation, 16 August 2006
- Carapezza M., Nuccio P.M., Valenza M., (1981) Genesis and Evolution of the fumaroles of Vulcano (Aeolian Islands, Italy): a geochemical model. *Bull. Volcanol.*, 44, 547-563
- Delemelle, P. & Stix, J., (2000) Volcanic Gases. In: eds. H. Sigurdsson, B. Houghton, S.R. McNutt, . Rymer & J. Stix, *Encyclopedia of Volcanoes*
- Le Guern F., Biccocchi P., Nihl A., Tazieff H., (1979) Analyse directe des gaz volcaniques. *C.R. Acad. Sc. Paris*, 288, série D, 867-870
- Obenholzner J.H., Glindemann D., Parks J., Edwards M., (2006) Filtering volcanic gases: Non-random, heterogeneous nucleation on single fibers, chemical composition of adsorber fluids and a link to the hydrogen economy. *Geophysical Research Abstract*, vol. 8, 05721
- Schorlemmer D., Wyss A., Maraini S., Wiemer S., Baer M., (2004) QuakeML – An XML Schema for seismology, *Orfeus Newsletter*, 6
- Sicardi L., (1941) Il recente ciclo dell'attività fumarolica dell'isola di Vulcano. *Bull. Volcanol.*, 7, 85-140
- Symonds, R.B., Rose, W.I., Bluth, G., Gerlach, T.M., (1994) Volcanic gas studies: methods, results, and applications, in Carroll, M.R., and Holloway, J.R., eds., Volatiles in Magmas: Mineralogical Society of America Reviews in *Mineralogy*, 30, pp 1-66
- Wood, L., Apparao, V., Champion, M., Hors, A. L., Pixley, T., Robie, J., Sharpe, P., and Wilson, C., (1999) Document Object Model (DOM) Level 2 Specification Version 1.0, *Working draft*, W3C

13. APPENDIX 1: VOLCANOGASML SCHEMA

```
<?xml version="1.0" encoding="utf-8"?>
<xd:schema xmlns:xd="http://www.w3.org/2001/XMLSchema">

<!-- ..... Element definitions ..... -->
  <xd:element name="VolcanoGasML">
    <xd:complexType>
      <xd:sequence>
        <xd:element ref="sample" minOccurs="1"/>
      </xd:sequence>
    </xd:complexType>
  </xd:element>
  <xd:element name="sample">
    <xd:complexType>
      <xd:sequence>
        <xd:element ref="location" minOccurs="1"/>
        <xd:element ref="chemical" minOccurs="1"/>
      </xd:sequence>
    </xd:complexType>
  </xd:element>
</xd:schema>
```

```

        </xd:sequence>
    </xd:complexType>
</xd:element>
<xd:element name="location">
    <xd:complexType>
        <xd:sequence>
            <xd:choice " minOccurs="1" maxOccurs="unbounded">
                <xd:element ref="volcano_name"/>
                <xd:element ref="gps_volcano"/>
            </xd:choice>
            <xd:element ref="country" minOccurs="0" maxOccurs="1"/>
            <xd:element ref="sample_number" minOccurs="0" maxOccurs="1"/>
            <xd:element ref="sample_date" minOccurs="0" maxOccurs="1"/>
            <xd:element ref="sample_location" minOccurs="0" maxOccurs="1"/>
            <xd:element ref="gps_location" minOccurs="0" maxOccurs="1"/>
            <xd:element ref="sampling_method" minOccurs="0"
maxOccurs="1"/>
        </xd:sequence>
    </xd:complexType>
</xd:element>

<xd:element name="chemical">
    <xd:complexType>
        <xd:sequence>
            <xd:element ref="T" minOccurs="0" maxOccurs="1"/>
            <xd:element ref="pH" minOccurs="0" maxOccurs="1"/>
            <xd:element ref="major_elements" minOccurs="0" maxOccurs="1"/>
            <xd:element ref="minor_elements" minOccurs="0" maxOccurs="1"/>
            <xd:element ref="isotopes" minOccurs="0" maxOccurs="1"/>
            <xd:element ref="ratio" minOccurs="0" maxOccurs="1"/>
        </xd:sequence>
    </xd:complexType>
</xd:element>
<xd:element name="volcano_name" mixed="true"/>
<xd:element name="country" mixed="true"/>
<xd:element name="sample_number" mixed="true"/>
<xd:element name="sample_date" mixed="true"/>
<xd:element name="sample_location" mixed="true"/>
<xd:element name="gps_location">
    <xd:complexType>
        <xd:sequence>
            <xd:element ref="latitude" minOccurs="1" maxOccurs="1"/>
            <xd:element ref="longitude" minOccurs="1" maxOccurs="1"/>
        </xd:sequence>
    </xd:complexType>
</xd:element>
<xd:element name="sampling_method" mixed="true"/>
<xd:element name="latitude" mixed="true"/>
<xd:element name="longitude" mixed="true"/>
<xd:element name="T">
    <xd:attribute name="TYPE" default="C">
        <xd:simpleType>
            <xd:restriction base="xd:string">
                <xd:enumeration value="C"/>
                <xd:enumeration value="F"/>
            </xd:restriction>
        </xd:simpleType>
    </xd:attribute>
</xd:element>

```



```

        </xd:restriction>
      </xd:simpleType>
    </xd:attribute>
  </xd:element>
  <xd:element name="pH" mixed="true"/>
  <xd:element name="major_elements">
    <xd:complexType>
      <xd:sequence>
        <xd:element ref="h2o" minOccurs="0" maxOccurs="1"/>
        <xd:element ref="so2" minOccurs="0" maxOccurs="1"/>
        <xd:element ref="co2" minOccurs="0" maxOccurs="1"/>
        <xd:element ref="hcl" minOccurs="0" maxOccurs="1"/>
        <xd:element ref="hf" minOccurs="0" maxOccurs="1"/>
        <xd:element ref="h2" minOccurs="0" maxOccurs="1"/>
        <xd:element ref="co" minOccurs="0" maxOccurs="1"/>
        <xd:element ref="so2" minOccurs="0" maxOccurs="1"/>
        <xd:element ref="s" minOccurs="0" maxOccurs="1"/>
        <xd:element ref="o2" minOccurs="0" maxOccurs="1"/>
      </xd:sequence>
    </xd:complexType>
  </xd:element>
  <xd:element name="h2o">
    <xd:complexType mixed="true">
      <xd:attribute name="unit" type="xd:string" use="required"/>
    </xd:complexType>
  </xd:element>
  <xd:element name="so2">
    <xd:complexType mixed="true">
      <xd:attribute name="unit" type="xd:string" use="required"/>
    </xd:complexType>
  </xd:element>
  <xd:element name="co2">
    <xd:complexType mixed="true">
      <xd:attribute name="unit" type="xd:string" use="required"/>
    </xd:complexType>
  </xd:element>
  <xd:element name="hcl">
    <xd:complexType mixed="true">
      <xd:attribute name="unit" type="xd:string" use="required"/>
    </xd:complexType>
  </xd:element>
  <xd:element name="hf">
    <xd:complexType mixed="true">
      <xd:attribute name="unit" type="xd:string" use="required"/>
    </xd:complexType>
  </xd:element>
  <xd:element name="h2">
    <xd:complexType mixed="true">
      <xd:attribute name="unit" type="xd:string" use="required"/>
    </xd:complexType>
  </xd:element>
  <xd:element name="co">
    <xd:complexType mixed="true">
      <xd:attribute name="unit" type="xd:string" use="required"/>
    </xd:complexType>
  </xd:element>

```

```
<xd:element name="so2">
  <xd:complexType mixed="true">
    <xd:attribute name="unit" type="xd:string" use="required"/>
  </xd:complexType>
</xd:element>
<xd:element name="s">
  <xd:complexType mixed="true">
    <xd:attribute name="unit" type="xd:string" use="required"/>
  </xd:complexType>
</xd:element>
<xd:element name="h2">
  <xd:complexType mixed="true">
    <xd:attribute name="unit" type="xd:string" use="required"/>
  </xd:complexType>
</xd:element>
<xd:element name="o2">
  <xd:complexType mixed="true">
    <xd:attribute name="unit" type="xd:string" use="required"/>
  </xd:complexType>
</xd:element>
<xd:element name="minor_elements">
  <xd:complexType>
    <xd:sequence>
      <xd:element ref="he" minOccurs="0" maxOccurs="1"/>
      <xd:element ref="ar" minOccurs="0" maxOccurs="1"/>
      <xd:element ref="ch4" minOccurs="0" maxOccurs="1"/>
      <xd:element ref="hbr" minOccurs="0" maxOccurs="1"/>
      <xd:element ref="nh3" minOccurs="0" maxOccurs="1"/>
      <xd:element ref="br" minOccurs="0" maxOccurs="1"/>
      <xd:element ref="ba" minOccurs="0" maxOccurs="1"/>
      <xd:element ref="b" minOccurs="0" maxOccurs="1"/>
      <xd:element ref="ph3" minOccurs="0" maxOccurs="1"/>
    </xd:sequence>
  </xd:complexType>
</xd:element>
<xd:element name="he">
  <xd:complexType mixed="true">
    <xd:attribute name="unit" type="xd:string" use="required"/>
  </xd:complexType>
</xd:element>
<xd:element name="ar">
  <xd:complexType mixed="true">
    <xd:attribute name="unit" type="xd:string" use="required"/>
  </xd:complexType>
</xd:element>
<xd:element name="ch4">
  <xd:complexType mixed="true">
    <xd:attribute name="unit" type="xd:string" use="required"/>
  </xd:complexType>
</xd:element>
<xd:element name="hbr">
  <xd:complexType mixed="true">
    <xd:attribute name="unit" type="xd:string" use="required"/>
  </xd:complexType>
</xd:element>
```

```
<xd:element name="nh3">
  <xd:complexType mixed="true">
    <xd:attribute name="unit" type="xd:string" use="required"/>
  </xd:complexType>
</xd:element>
<xd:element name="br">
  <xd:complexType mixed="true">
    <xd:attribute name="unit" type="xd:string" use="required"/>
  </xd:complexType>
</xd:element>
<xd:element name="ba">
  <xd:complexType mixed="true">
    <xd:attribute name="unit" type="xd:string" use="required"/>
  </xd:complexType>
</xd:element>
<xd:element name="b">
  <xd:complexType mixed="true">
    <xd:attribute name="unit" type="xd:string" use="required"/>
  </xd:complexType>
</xd:element>
<xd:element name="ph3">
  <xd:complexType mixed="true">
    <xd:attribute name="unit" type="xd:string" use="required"/>
  </xd:complexType>
</xd:element>
<xd:element name="isotopes">
  <xd:complexType>
    <xd:sequence>
      <xd:element ref="222Rn" minOccurs="0" maxOccurs="1"/>
      <xd:element ref="220Rn" minOccurs="0" maxOccurs="1"/>
    </xd:sequence>
  </xd:complexType>
</xd:element>
<xd:element name="222Rn">
  <xd:complexType mixed="true">
    <xd:attribute name="unit" type="xd:string" use="required"/>
  </xd:complexType>
</xd:element>
<xd:element name="220Rn">
  <xd:complexType mixed="true">
    <xd:attribute name="unit" type="xd:string" use="required"/>
  </xd:complexType>
</xd:element>
<xd:element name="ratio">
  <xd:complexType>
    <xd:sequence>
      <xd:element ref="gas-stream" minOccurs="1" maxOccurs="1"/>
    </xd:sequence>
  </xd:complexType>
</xd:element>
<xd:element name="gas-stream" mixed="true"/>
</xd:schema>
```

14. APPENDIX 2 : AN EXAMPLE OF VOLCANOGASML FILE

Data used for this example are from Sicardi (1941), Carapezza et al. (1981) and Le Guern et al. (1979).

```
<VolcanoGasML>
  <sample>
    <location>
      <volcano_name>Vulcano</volcano_name>
      <country>Southern Italy</country>
      <sample_date>1923</sample_date>
    </location>
    <chemical>
      <T>480</T>
      <major_elements>
        <h2o unit="% vol.">96.5</h2o>
        <so2 unit="% vol.">21.2</so2>
        <co2 unit="% vol.">55.2</co2>
        <hcl unit="% vol.">1.2</hcl>
      </major_elements>
    </chemical>
  </sample>
  <sample>
    <location>
      <volcano_name>Vulcano</volcano_name>
      <country>Southern Italy</country>
      <sample_date>1978</sample_date>
    </location>
    <chemical>
      <T>281</T>
      <major_elements>
        <h2o unit="% vol.">81.9</h2o>
        <so2 unit="% vol.">5.2</so2>
        <co2 unit="% vol.">93.9</co2>
      </major_elements>
    </chemical>
  </sample>
  <sample>
    <location>
      <volcano_name>Vulcano</volcano_name>
      <country>Southern Italy</country>
      <sample_date>1978</sample_date>
    </location>
    <chemical>
      <T>220</T>
      <major_elements>
        <so2 unit="% vol.">95.3</so2>
        <co2 unit="% vol.">93.9</co2>
      </major_elements>
    </chemical>
  </sample>
</VolcanoGasML>
```