

Network Working Group
Request for Comments: 2077
Category: Standards Track

S. Nelson
LLNL
C. Parks
NIST
Mitra
WorldMaker
January 1997

The Model Primary Content Type for Multipurpose Internet Mail Extensions

Status of this Memo

This document specifies an Internet standards track protocol for the Internet community, and requests discussion and suggestions for improvements. Please refer to the current edition of the "Internet Official Protocol Standards" (STD 1) for the standardization state and status of this protocol. Distribution of this memo is unlimited.

Introduction

The purpose of this memo is to propose an update to Internet RFC 2045 to include a new primary content-type to be known as "model". RFC 2045 [1] describes mechanisms for specifying and describing the format of Internet Message Bodies via content-type/subtype pairs. We believe that "model" defines a fundamental type of content with unique presentational, hardware, and processing aspects. Various subtypes of this primary type are immediately anticipated but will be covered under separate documents.

Table of Contents

1. Overview.....	2
2. Definition.....	2
3. Consultation Mechanisms.....	4
4. Encoding and Transport.....	5
5. Security Considerations Section.....	6
6. Authors' Addresses.....	7
7. Expected subtypes.....	7
8. Appendix.....	9
9. Acknowledgements.....	13

1. Overview

This document will outline what a model is, show examples of models, and discuss the benefits of grouping models together. This document will not directly deal with the intended subtypes since those will be covered by their separate registrations. Some immediately expected subtypes are listed in section 7.

This document is a discussion document for an agreed definition, intended eventually to form a standard accepted extension to RFC 2045. We are also targeting developers of input/output filters, viewer software and hardware, those involved in MIME transport, and decoders.

2. Definition of a model

A model primary MIME type is an electronically exchangeable behavioral or physical representation within a given domain. Each subtype in the model structure has unique features, just as does each subtype in the other primary types. The important fact is that these various subtypes can be converted between each other with less loss of information than to that of other primary types. This fact groups these subtypes together into the model primary type. All of the expected subtypes have several features in common and that are unique to this primary type.

To loosely summarize: models are multidimensional structures composed of one or more objects. If there are multiple objects then one object defines the arrangement/setting/relationship of the others. These objects all have calibrated coordinate systems but these systems need not be in the same units nor need they have the same dimensionality. In detail:

1. have 3 or more dimensions which are bases of the system and form an orthogonal system (any orthogonal system is sufficient).

This system is specifically defined in terms of an orthogonal set of basis functions [for a subspace of the L^2 function space] over a coordinate system of dimension 3 or more. Note that this does not preclude regular skewed systems, elliptical coordinates, different vector spaces, etc.

2. contain a structural relationship between model elements.
3. have scaling or calibration factors which are related to physical units (force, momentum, time, velocity, acceleration, size, etc.). Thus, an IGES file will specify a building of non-arbitrary size, computational meshes and VRML models will have real spatial/

temporal units. This allows for differing elements to be combined non-arbitrarily.

4. Models can be single objects or composed of a collection of objects. These normally independent objects are arranged in a master/slave scenario so that one object acts as the reference, or primary object, which defines how the other objects interrelate and behave. This allows for the creation of mathematical, physical, economic, behavioral, etc. models which typically are composed of different elements. The key is in the description: these types describe how something "behaves"; contrasted to typical data types which describe how something "is".

The inclusion of this "collective" system works similar to the Email system's multipart/related type which defines the actions of the individual parts. Further specification of the model/* subtypes utilizing these properties is left to the subtype authors.

With these assumptions:

- a. the default dimensionality will be spatial and temporal (but any are allowed).
- b. it is presumed that models will contain underlying structure which may or may not be immediately available to the user. (fluid dynamics vector fields, electromagnetic propagation, interrelated IGES dimensional specifiers, VRML materials and operators, etc.)
- c. it is assumed that basis set conversion between model domains is lossless. The interpretation of the data may change but the specification will not. i.e. convert the model of the U.S.A. Gross Domestic Product into a VRML model and navigate it to explore the variances and interrelationships. The model has many dimensions but also "passages" and "corridors" linking different parts of it. A similar situation is true for meshes and CAD files. The key is identifying the basis set conversion which makes sense.
- d. models are grouped to assure LESS loss of information between the model subtypes than to subtypes of other primary types. (i.e. converting a chemical model into an image is more lossy than concerting it into a VRML model).

Items c and d above define the grouping for model similar to the way that "images" and "videos" are grouped together; to assure less loss of information. Obviously converting from a GIF image to a JPEG image loses less information than converting from a GIF image to an AU audio file.

3. Consultation Mechanisms

Before proposing a subtype for the model/* primary type, it is suggested that the subtype author examine the definition (above) of what a model/* is and the listing (below) of what a model/* is not. Additional consultations with the authors of the existing model/* subtypes is also suggested.

Copies of RFCs are available on:

<ftp://ftp.isi.edu/in-notes/>

Copies of Internet-Drafts are available on:

<ftp://ftp.ietf.org/internet-drafts/>

Similarly, the VRML discussion list has been archived as:

<http://vrml.wired.com/arch/>

and discussions on the comp.mail.mime group may be of interest. Discussion digests for the existing model/* subtypes may be referenced in the respective documents.

The mesh community presently has numerous different mesh geometries as part of different packages. Freely available libraries need to be advertised more than they have been in the past to spur the development of interoperable packages. It is hoped that by following the example of the VRML community and creating a freely available comprehensive library of input/output functions for meshes [11] that this problem will be alleviated for the mesh community. A freely available mesh viewer conforming to these standards is available now for various platforms. Consultations with the authors of the mesh system,

<http://www-dsed.llnl.gov/documents/tests/mesh.html>

will be beneficial.

The IGES community has a suite of tests and conformance utilities to gauge the conformance to specifications and software authors are encouraged to seek those out from NIST [14].

4. Encoding and Transport

- a. Unrecognized subtypes of model should at a minimum be treated as "application/octet-stream". Implementations may optionally elect to pass subtypes of model that they do not specifically recognize to a robust general-purpose model viewing application, if such an application is available.
- b. Different subtypes of model may be encoded as textual representations or as binary data. Unless noted in the subtype registration, subtypes of model should be assumed to contain binary data, implying a content encoding of base64 for email and binary transfer for ftp and http.
- c. The formal syntax for the subtypes of the model primary type should look like this:

Media type name:	model
Media subtype name:	xxxxxxx
Required parameters:	none
Optional parameters:	dimensionality, state (see below)
Encoding considerations:	base64 encoding is recommended when transmitting model/* documents through MIME electronic mail.
Security considerations:	see section 5 below
Published specification:	This document. See Appendix B for references to some of the expected subtypes.
Person and email address to contact for further information:	Scott D. Nelson <nelson18@llnl.gov> 7000 East Ave. Lawrence Livermore National Laboratory Livermore, CA 94550

The optional parameters consist of starting conditions and variable values used as part of the subtypes. A base set is listed here for illustration purposes only and will be covered in detail as part of the respective subtypes:

dimension := string ; a number indicating the number of dimensions.
This is used as a "hint" in selecting applicable viewer programs.

state := string ; "static" or "dynamic". In "static", the observer may move about, thus effecting translations, rotations, pans, zooms, etc. but the data does not change. In "dynamic", the data itself is manipulated via skews, elongations, scales, etc. Note that time evolution is still a static operation since it is just a translation along one of the principal dimensions while the elongation of a cube or object deformation are dynamic operations.

Note that this optional parameter list does not limit those specified by the various subtypes.

- d. The specific issues relating to the various subtypes are covered as part of the description of those specific subtypes. The following is an example of a typical MIME header used for mail transport purposes:

```
To:    you@some.org
From:  nelson18@llnl.gov
Date:  Fri, 30 Aug 96 13:33:19 -0700
Content-Type: model/mesh; dimension="4"; state="static"
Content-Transfer-Encoding: base64
MIME-Version: 1.0
Subject: model data file

I1ZSTUwgVjEuMCBhc2NpaQojIFRoXMGZmlsZSB3YXMgIGdlbmVyY...
byBDb21tdW5pY2F0aW9ucwojIGh0dHA6Ly93d3cuY2hhY28uY29tC...
IyB1c2VkIGluIHJvb20gMTkyICh0ZXN0IHJvb20pCiAgIAojIFRvc...
.
.
.
```

5. Security Considerations Section

Note that the data files are "read-only" and do not contain file system modifiers or batch/macro commands. The transported data is not self-modifying but may contain interrelationships. The data files may however contain a "default view" which is added by the author at file creation time. This "default view" may manipulate viewer variables, default look angle, lighting, visualization options, etc. This visualization may also involve the computation of variables or values for display based on the given raw data. For motorized equipment, this may change the position from the hardware's rest state to the object's starting orientation.

The internal structure of the data files may direct agents to access additional data from the network (i.e. inclusions); the security limits of whom are not pre-supposed. Actions based on these inclusions are left to the security definitions of the inclusions. Further comments about the security considerations for the subtypes will be contained in each subtype's registration.

6. Authors' Addresses

S. D. Nelson
Lawrence Livermore National Laboratory,
7000 East Ave., L-153,
Livermore CA 94550, USA.
E-Mail: nelson18@llnl.gov

C. Parks
National Institute of Standards & Technology
Bldg 220, Room B-344
Gaithersburg, MD 20899, USA.
E-Mail: parks@eeel.nist.gov

Mitra
WorldMaker
1056 Noe
San Francisco, CA 94114
E-Mail: mitra@earth.path.net

7. Expected subtypes

Table 1 lists some of the expected model sub-type names. Suggested 3 letter extensions are also provided for DOS compatibility but their need is hopefully diminished by the use of more robust operating systems on PC platforms. The "silo" extension is provided for backwards compatibility. Mesh has an extensive list of hints since the present variability is so great. In the future, the need for these hints will diminish since the files are self describing. This document is not registering these subtypes. They will be handled under separate documents.

Table 1.

Primary/sub-type	Suggested extension(s)	Reference
model/iges	igs, iges	[8]
model/vrml	wrl	[9]
model/mesh	msh, mesh, silo	[10]

It is expected that model/mesh will also make use of a number of parameters which will help the end user determine the data type without examine the data. However, note that mesh files are self-describing.

regular+static, unstructed+static, unstructured+dynamic,
conformal+static, conformal+dynamic, isoparametric+static,
isoparametric+dynamic

The sub-types listed above are some of the anticipated types that are already in use. Notice that the IGES type is already registered as "application/iges" and that RFC states that a more appropriate type is desired. Note that the author of "application/iges" is one of the authors of this "model" submission and application/iges will be re-registered as model/iges at the appropriate time.

The VRML type is gaining wide acceptance and has numerous parallel development efforts for different platforms. These efforts are fueled by the release of the QvLib library for reading VRML files; without which the VRML effort would be less further along. This has allowed for a consistent data type and has by defacto established a set of standards. Further VRML efforts include interfaces to other kinds of hardware (beyond just visual displays) and it is proposed by those involved in the VRML effort to encompass more of the five senses. Unlike other kinds of "reality modeling" schemes, VRML is not proprietary to any one vendor and should experience similar growth as do other open standards.

The mesh type is an offshoot of existing computational meshing efforts and, like VRML, builds on a freely available library set. Also like VRML, there are other proprietary meshing systems but there are converters which will convert from those closed systems to the mesh type. Meshes in general have an association feature so that the connectivity between nodes is maintained. It should be noted that most modern meshes are derived from CAD solids files.

8. Appendices

8.1 Appendix A -- extraneous details about expected subtypes

VRML Data Types

The 3D modeling and CAD communities use a number of file formats to represent 3D models, these formats are widely used to exchange information, and full, or lossy, converters between the formats exist both independently and integrated into widely used applications. The VRML format is rapidly becoming a standard for the display of 3D information on the WWW.

Mesh Data Types

For many decades, finite element and finite difference time domain codes have generated mesh structures which attempt to use the physical geometry of the structures in connection with various physics packages to generate real world simulations of events including electromagnetic wave propagation, fluid dynamics, motor design, etc. The resulting output data is then post processed to examine the results in a variety of forms. This proposed mesh subtype will include both geometry and scalar/vector/tensor results data. An important point to note is that many modern meshes are generated from solids constructed using CAD packages.

Motivation for mesh grew out of discussions with other communities about their design requirements. Many CAD or scene descriptions are composed of a small number of complex objects while computational meshes are composed of large numbers of simple objects. A 1,000,000 element 3D mesh is small. A 100,000,000 element 3D structured mesh is large. Each object can also have an arbitrary amount of associated data and the mesh connectivity information is important in optimizing usage of the mesh. Also, the mesh itself is usually uninteresting but postprocessing packages may act on the underlying data or a computational engine may process the data as input.

Meshes differ principally from other kinds of scenes in that meshes are composed of a large number of simple objects which may contain arbitrary non-spatial parameters, not all of whom need be visible, and who have an implicit connectivity and neighbor list. This latter point is the key feature of a mesh. It should be noted that most meshes are generated from CAD files however. The mesh type has association functions because the underlying physics was used to calculate the interaction (if you crash a car into a telephone pole, you get a crumpled car and a bent pole). Most interesting computational meshes are 4D with additional multidimensional results components.

IGES CAD Data Types

(The following text, reproduced for reference purposes only, is from "U.S. Product Data Association and IGES/PDES Organization Reference Manual," June 1995; by permission.)

IGES, the Initial Graphics Exchange Specification, defines a neutral data format that allows for the digital exchange of information among computer-aided design (CAD) systems.

CAD systems are in use today in increasing numbers for applications in all phases of the design, analysis, and manufacture and testing of products. Since the designer may use one supplier's system while the contractor and subcontractor may use other systems, there is a need to be able to exchange data digitally among all CAD systems.

The databases of CAD systems from different vendors often represent the same CAD constructs differently. A circular arc on one system may be defined by a center point, its starting point and end point, while on another it is defined by its center, its diameter starting and ending angle. IGES enables the exchange of such data by providing, in the public domain, a neutral definition and format for the exchange of such data.

Using IGES, the user can exchange product data models in the form of wireframe, surface, or solid representations as well as surface representations. Translators convert a vendor's proprietary internal database format into the neutral IGES format and from the IGES format into another vendor's internal database. The translators, called pre- and post-processors, are usually available from vendors as part of their product lines.

Applications supported by IGES include traditional engineering drawings as well as models for analysis and/or various manufacturing functions. In addition to the general specification, IGES also includes application protocols in which the standard is interpreted to meet discipline specific requirements.

IGES technology assumes that a person is available on the receiving end to interpret the meaning of the product model data. For instance, a person is needed to determine how many holes are in the part because the hole itself is not defined. It is represented in IGES by its component geometry and therefore, is indistinguishable from the circular edges of a rod.

The IGES format has been registered with the Internet Assigned Numbers Authority (IANA) as a Multipurpose Internet Mail Extension (MIME) type "application/iges". The use of the message type/subtype

in Internet messages facilitates the uniform recognition of an IGES file for routing to a viewer or translator.

Version 1.0 of the specification was adopted as an American National Standards (ANS Y14.26M-1981) in November of 1981. Versions 3.0 and 4.0 of the specification have subsequently been approved by ANSI. The current version of IGES 5.2 was approved by ANSI under the new guidelines of the U.S. Product Data Association. Under these guidelines, the IGES/PDES Organization (IPO) became the accredited standards body for product data exchange standards. This latest standard is USPRO/IPO-100-1993.

8.2 Appendix B -- References and Citations

[1] Freed, N., and N. Borenstein, "Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies", RFC 2045, Innosoft, First Virtual, November 1996.

[2] Fitzgerald P., "Molecules-R-Us Interface to the Brookhaven Data Base", Computational Molecular Biology Section, National Institutes of Health, USA; see <http://www.nih.gov/htbin/pdb> for further details; Peitsch M.C, Wells T.N.C., Stampf D.R., Sussman S. J., "The Swiss-3D Image Collection And PDP-Browser On The Worldwide Web", Trends In Biochemical Sciences, 1995, 20, 82.

[3] "Proceedings of the First Electronic Computational Chemistry Conference", Eds. Bachrach, S. M., Boyd D. B., Gray S. K, Hase W., Rzepa H.S, ARInternet: Landover, Nov. 7- Dec. 2, 1994, in press; Bachrach S. M, J. Chem. Inf. Comp. Sci., 1995, in press.

[4] Richardson D.C., and Richardson J.S., Protein Science, 1992, 1, 3; D. C. Richardson D. C., and Richardson J.S., Trends in Biochem. Sci., 1994, 19, 135.

[5] Rzepa H. S., Whitaker B. J., and Winter M. J., "Chemical Applications of the World-Wide-Web", J. Chem. Soc., Chem. Commun., 1994, 1907; Casher O., Chandramohan G., Hargreaves M., Murray-Rust P., Sayle R., Rzepa H.S., and Whitaker B. J., "Hyperactive Molecules and the World-Wide-Web Information System", J. Chem. Soc., Perkin Trans 2, 1995, 7; Baggott J., "Biochemistry On The Web", Chemical & Engineering News, 1995, 73, 36; Schwartz A.T, Bunce D.M, Silberman R.G, Stanitski C.L, Stratton W.J, Zipp A.P, "Chemistry In Context - Weaving The Web", Journal Of Chemical Education, 1994, 71, 1041.

[6] Rzepa H.S., "WWW94 Chemistry Workshop", Computer Networks and ISDN Systems, 1994, 27, 317 and 328.

- [7] S.D. Nelson, "Email MIME test page", Lawrence Livermore National Laboratory, 1994. See <http://www-dsed.llnl.gov/documents/WWWtest.html> and <http://www-dsed.llnl.gov/documents/tests/email.html>
- [8] C. Parks, "Registration of new Media Type application/iges", <ftp://ftp.isi.edu/in-notes/iana/assignments/media-types/application/iges>, 1995.
- [9] G. Bell, A. Parisi, M. Pesce, "The Virtual Reality Modeling Language", <http://sdsc.edu/SDSC/Partners/vrml/Archives/vrml10-3.html>, 1995.
- [10] S.D. Nelson, "Registration of new Media Type model/mesh", <ftp://ftp.isi.edu/in-notes/iana/assignments/media-types/model/mesh>, 1997.
- [11] "SILO User's Guide", Lawrence Livermore National Laboratory, University of California, UCRL-MA-118751, March 7, 1995,
- [12] E. Brugger, "Mesh-TV: a graphical analysis tool", Lawrence Livermore National Laboratory, University of California, UCRL-TB-115079-8, http://www.llnl.gov/liv_comp/meshtv/mesh.html
- [13] S. Brown, "Portable Application Code Toolkit (PACT)", the printed documentation is accessible from the PACT Home Page http://www.llnl.gov/def_sci/pact/pact_homepage.html
- [14] L. Rosenthal, "Initial Graphics Exchange Specification (IGES) Test Service", <http://speckle.ncsl.nist.gov/~jacki/igests.htm>

8.3 Appendix C -- hardware

Numerous kinds of hardware already exist which can process some of the expected model data types and are listed here for illustration purposes only:

stereo glasses, 3D lithography machines, automated manufacturing systems, data gloves (with feedback), milling machines, aromascopes, treadmills.

8.4 Appendix D -- Examples

This section contains a collection of various pointers to examples of what the model type encompasses:

Example mesh model objects can be found on this mesh page:
<http://www-dsed.llnl.gov/documents/tests/mesh.html>

Various IGES compliant test objects:
<http://www.eeel.nist.gov/iges/specfigures/index.html>

VRML Test Suite:
<http://www.chaco.com/vrml/test/>

An image of a model of a shipping cage crashing into the ground:
http://www.llnl.gov/liv_comp/meiko/apps/dyna3d/cagefig2.gif

An image of a 100,000,000 zone mesh:
http://www.llnl.gov/liv_comp/meiko/apps/hardin/PMESH.gif

A video of a seismic wave propagation through a computational mesh:
http://www.llnl.gov/liv_comp/meiko/apps/larsen/movie.mpg

9. Acknowledgements

Thanks go to Henry Rzepa (h.rzepa@ic.ac.uk), Peter Murray-Rust (pmr1716@ggr.co.uk), Benjamin Whitaker (B.J.Whitaker@chemistry.leeds.ac.uk), Bill Ross (ross@cgl.ucsf.EDU), and others in the chemical community on which the initial draft of this document is based. That document updated an IETF Internet Draft in which the initial chemical submission was made, incorporated suggestions received during the subsequent discussion period, and indicated scientific support for and uptake of a higher level document incorporating physical sciences[2-7]. This Model submission benefited greatly from the previous groundwork laid, and the continued interest by, those communities.

The authors would additionally like to thank Keith Moore (moore@cs.utk.edu), lilley (lilley@afs.mcc.ac.uk), Wilson Ross (ross@cgl.ucsf.EDU), hansen (hansen@pegasus.att.com), Alfred Gilman (asg@severn.wash.inmet.com), and Jan Hardenbergh (jch@nell.oki.com) without which this document would not have been possible. Additional thanks go to Mark Crispin (MRC@CAC.Washington.EDU) for his comments on the previous version and Cynthia Clark (cclark@ietf.org) for editing the submitted versions.