ELSEVIER

Multiple sequence alignment in HTML: Colored, possibly hyperlinked, compact representations*

F. Campagne and B. Maigret

Laboratoire de Chimie théorique, Université Henri Poincaré Nancy I, UMR CNRS/UHP 7565, Vandoeuvre-les-Nancy, France

Protein sequence alignments are widely used in protein structure prediction, protein engineering, modeling of proteins, etc. This type of representation is useful at different stages of scientific activity: looking at previous results, working on a research project, and presenting the results. There is a need to make it available through a network (intranet or WWW), in a way that allows biologists, chemists, and noncomputer specialists to look at the data and carry on research—possibly in a collaborative research. Previous methods (text-based, Java-based) are reported and their advantages are discussed. We have developed two novel approaches to represent the alignments as colored, hyper-linked HTML pages. The first method creates an HTML page that uses efficiently the image cache mechanism of a WWW browser, thereby allowing the user to browse different alignments without waiting for the images to be loaded through the network, but only for the first viewed alignment. The generated pages can be browsed with any HTML2.0-compliant browser. The second method that we propose uses W3C-CSS1-style sheets to render alignments. This new method generates pages that require recent browsers to be viewed. We implemented these methods in the Viseur program and made a WWW service available that allows a user to convert an MSF alignment file in HTML for WWW publishing. The latter service is available at http://www.lctn.u-nancy.fr/viseur/services.html. © *1998 by Elsevier Science Inc.*

Keywords: multiple sequence alignment, HTML representation

Color Plates for this article are on pages 34 and 35.

Address reprint requests to: B. Maigret, Laboratoire de Chimie théorique, Université Henri Poincaré Nancy I, UMR CNRS/UHP 7565, BP 239, 54506 Vandoeuvre-les-Nancy, France. e-mail: campagne@lctn.u-nancy.fr; maigret@lctn.u-nancy.fr.

*Paper submitted to Electronic Conference of the Molecular Graphics and Modelling Society, MGM EC-2, October 1997.

Received 6 January 1998; revised 5 April 1998; accepted 16 April 1998.

INTRODUCTION

General interest in sequence alignment representations

Multiple sequence alignments are useful for comparing, working with, and representing related (bio)polymer sequences. They help to highlight similarities and differences between regions of sequences that would not be apparent when considering sequences individually. Program tools[1] have been designed to present and display multiple sequence alignments (MSAs) to assist in publication, etc., on paper media. We present here methods to achieve these results with network media: intranets and the WWW.

Interest in MSA network representations

There is a growing interest in accessing all kinds of research results through a network. Protein sequence-oriented data are not an exception to this trend: from database content to research results, people would benefit from MSAs being integrated with other materials. We observed several examples of this while working on the Viseur project,[2] and while collaborating on GPCRDB.[3]

Visualization of MSAs is also important for collaborative research, and people involved in collaborations know how much improvements in efficiency can be obtained using network tools. In such cases, people need to view, comment on partial results from collaborators, and create representations of their own results. As an MSA makes it easier to communicate key ideas about a set of sequences, it should not be neglected by the information exchange collaborative system used to carry on the collaborative task.

This explains the need for an appropriate representation of MSA data for use through a network. Our intention is to restrict ourselves to HTTP technology,[4] the common foundation for

655 Avenue of the Americas, New York, NY 10010

1093-3263/98/$19.00
PII S1093-3263(98)00010-2

WWW and intranet technologies. With respect to our background, working with MSAs and HTTP, an appropriate representation should be

- Portable: the representation should be available to the broadest audience. This means that documents have to be created that will be displayed in the same way by different browsers; or at least, that the differences in rendering the document will not alter the information published
- Colored: to highlight class or individual properties of residues
- Possible to hyperlink at the residue level: In the Viseur program we found it valuable to have some residues hyperlinked to mutation data
- Transferable to the user local working area: Local copy of the MSA on user's working machine or network for fast access or data compilation

From a server-side point of view, it requires that MSAs be represented in one of the following ways:

- HTML pages: text, images, etc.
- Helper applications of other document types: e.g., proprietary file formats
- Java applets

In the next section we describe how these features are used, or not used to represent MSAs on the WWW, and whether they achieve our criteria. The Methods section of this article describes two novel methods we propose to generate MSA representations that fit the constraints we have described.

Current representation of alignment data on intranet or WWW

Text-based HTML representation Text-based HTML representation is currently the most common mode of representation. Alignments are included in an HTML document as simple text, using a fixed-pitch font.[†] Vertical and horizontal alignments are found, but the user cannot switch according to his or her preference: the choice was made by the creator of the page (sometimes, the author proposes both orientations). For example, to represent an alignment of two peptidic sequences, the piece of HTML illustrated in Color Plate 1A can be used.

The second portion of the representation in Color Plate 1A displays the sequences as rendered by your browser. To this representation, it is possible to add colors: The Alignment Colour Viewer[5] is a WWW tool with which to create HTML alignment pages from two sequences (with a few options to change the coloring of the residues according to four sets of residues: Blue, Red, Green, Yellow). Color Plate 1B presents an example of this service output.

Using a similar text-based HTML, it is also possible to hyperlink some residues with information that refers to them. Consider, for example, the HTML fragment shown in Color Plate 1C.

Notice that colors and hyperlinks cannot be used independently: our intention was to color one block of amino acids red, but we had (at least on our browsers) two red blocks (P/I, GFT/GFS), separated by one block whose color depended on browser configuration, i.e., GG/GG. Hyperlinks may have a higher priority than colors for a particular browser, so that the GG pair would not be colored red (or it could be red, until the link is selected). Its color would then change to the selected link color. In fact, there is no standard convention that describes what happens to text colors when the text is also hyperlinked. It would be bad design to rely on the behavior of a particular browser to represent local properties of an MSA. Thus, we find that this representation can be used to represent sequence alignments with hyperlinks and without colors, or with colors but without hyperlinks.

External viewer representation External viewer representation was not used to represent sequence alignment. It would require an alignment viewer program to be installed on each client machine. The installation of the alignment viewer may be the source of problems, because

- Installation of an external viewer, using MIME (Multipurpose Internet Mail Extensions) conventions, although not difficult, requires experience that the end-user may not have
- An alignment viewer would have to be available, for every platform, reading the same file format. At the present time, it is not clear whether such a collection of tools exists; if it does exist; will it be maintained on each platform?

These considerations are less important for intranet publishing: in these situations, both the server and the client technology are under control. But for both the WWW and intranets, external viewer usage would make difficult the integration of alignment data with other materials, or to hyperlink residues of the alignments to information.[‡] For these reasons, we believe it is not an interesting MSA representation technique.

Java applet representation CINEMA[7,8] is an applet developed for visualization and edition of multiple alignments. To write this visualization tool, its authors chose the Java language, which ensures portability and broad availability for such software, at least in theory. Colored alignments can be displayed and changed graphically. Residue hyperlinking is not supported by the last release in our possession (version 2.02). Currently, only a few ways exist to transmit the alignment to the applet, which requires the alignment to be present on the WWW server machine in a specific format. CINEMA is widely, and freely, available to the community to create new local alignment servers.

Alignment files are not difficult to create, so that CINEMA could constitute a good MSA viewer when the following criteria are met:

- Portability could be limited to Java-enabled browers (this limits the audience to whom the information is provided)
- Annotations are not required on the page where the alignment is (CINEMA creates a new window for each alignment where annotations cannot be included)
- Complex page layout are not required (vertical alignments or small pieces of alignment comparisons may require complex page layout)
- Hyperlinks on residues are not needed.

[†]Each graphical representation of characters in fixed-pitch fonts spreads over the same width on the page, whatever the character is. For example, a `fixed-pitch` font.

[‡]PostScript MSA representation, for example,[6] suffers from these limitations.

We were not ready to accept these limitations and decided to develop methods to render MSAs using HTML.[§] These methods are presented in the remainder of this article.

METHODS

Although the methods are illustrated for protein MSAs, they are applicable to whole MSA representations.

Single image

A simple method to generate hyperlinked HTML alignments would be to create images such as that shown in Color Plate 1D. The DRY column is hyperlinked to information, via a standard map (client and server-side image maps). Unfortunately, when the alignment becomes larger, time transfer for images makes that method no longer useful in practice. Our illustration contains 265 positions (residues and gaps) for a size of 341 406 bytes (average, 118 bytes per position).

Small set of small images

To address this performance problem, we decided to take advantage of the fact that proteins are constituted of repeated units. We generated sequence alignment HTML pages made of small colored images. The HTML fragment shown in Color Plate 1E is constructed on the basis of this idea.

Each residue is assigned a picture that must be accessible from the name "images/AA?.gif." The question mark encodes a residue (? = {A, C, D, E, F, G, H, I, K, L, M, N, P, Q, R, S, T, V, W, Y}) or a gap (? = {_}).

This type of representation, to be viewed under good conditions, requires the user to have a graphical browser that renders efficiently many identical small images from its cache.

The real example shown in the original HTML article, illustrates that performance is better than with a single image. Our example contains 6 050 small images, which are loaded as 21 000 bytes (20 residues, one gap, less than 1 000 bytes each) for the images, plus the size of the HTML document itself (179 033 bytes). The total is 185 083 bytes (average, 31 bytes per position).

Using the simple image generation method would require a 760-kb image size: approximately four times larger.

Introducing hyperlinks Hyperlinks can be added to this representation in a way that is portable across browsers. To solve the portability problem owing to browsers overlining a hyperlinked image with a border, one can use the BORDER tag, conjugated with explicit WIDTH and HEIGHT tags. A few examples will help explain these precautions: Consider Color Plate 2A.

The border was set to 10 pixels, in order to make it clearly visible. This shows how a few hyperlinked residues can disturb the overall alignment if no precaution is taken: the border simply enlarges the residue position, and this offsets following positions by twice the border width. To correct this behavior in a portable way, it is not enough to reduce the size of hyperlinked images, by twice the border, in both dimensions. It is also necessary to use the BORDER tag, to force the browser to use the exact border value we used for the reduction.

The example in Color Plate 2B shows an appropriate piece of HTML alignment, including hyperlinks (first line). Last, this graphical representation can be proposed together with a different set of small residue images to provide for user customization (images of different sizes, different color schemes, etc.). One way of doing this consists in using different image directories and creating the HTML representation on the fly, according to user preferences.

Cascade style sheets

Cascade style sheets (CSS's) have been partially implemented in WWW graphical browsers. CSS1 recommendations[4] are not entirely supported by any browser, at the present time. Fortunately, a subset of CSS1 is enough to represent a multiple sequence alignment, and this subset is supported by the browsers most frequently used by experimenters or modelers: InternetExplorer 3.0+ and Netscape 4.0+.

For more information on using CSS's, the interested reader is encouraged to consult the Cascading Style Sheets Page,[9] at W3C. Here, we focus on CSS1 elements that are essential to the representation we propose. To represent the residue colors, we use the ability to change the background color of characters with the SPAN tag. This eliminates the need for colored images, which, surprisingly, are not efficiently rendered by style sheet-compliant browsers.

Into the HTML HEADER, new classes are added to the SPAN HTML tag, which encode the style for each residue. The style contains

- A restriction to monospaced character fonts, to ensure that each residue will be rendered with the same width, to conserve the alignment
- Encoding of the foreground color
- Encoding of the background color

The definition of SPAN classes is enclosed in comments. This is a common trick to avoid misinterpretation by non-CSS1-compliant browsers of style definitions. In the body of the document, each residue code is preceded by the SPAN tag, the class of which corresponds to the residue. Like common tags, the SPAN tag is closed when its effect is no longer required (⟨/SPAN⟩). Please note that when a few residue classes are to be distinguished, it may become interesting to factorize the SPAN tags, per residue class. For example: ⟨/SPAN CLASS = A⟩ALILLL⟨/SPAN⟩ colors contiguous aliphatic residues using the aliphatic class to color by the mean of the class.

Introducing hyperlinks

Color Plate 2C illustrates this method used with hyperlinks. Here, each residue is encoded as HTML text. To hyperlink text, browsers sometimes underline the text, and always change its foreground color to a user-defined color. Because we used the background to encode the property color, and because underlining, when it occurs, does not change the bounding box of characters, we are able to add hyperlinks without any other precautions.

[§]This should not be taken as a criticism of CINEMA: this package was clearly designed as a highly interactive MSA *editor* and this constraint makes it somewhat unadapted to the use we present here (i.e., as an MSA *view*).

RESULTS

Because the methods are near impossible to follow by hand, for real case alignments we implemented three methods in the Viseur program: (1) a text-based method, (2) a graphical method, and (3) a style-sheet method. Because this program was developed to help the modeling of integral membrane proteins (GPCR originally), it is limited to handle MSAs of proteins. These implementations enable us to provide a WWW service for translating MSF sequence alignments to HTML pages. This service is available free of charge, through the WWW, from our server. Presented here are illustrations of this service output, for the same MSA, using HTML alignments produced by

- A text-based method
- A graphical method
- A style-sheet method

The current release of this service does not allow generation of alignments with hyperlinked residues; but the program itself does. Should this feature be needed, we suggest that the Viseur program be obtained. Those interested in using the conversion methods intensively, or who prefer not to submit confidential alignments through the WWW, are encouraged to obtain and install the Viseur program, for their private use on their own machine. More information on this program, including the description of our distribution policy, is available from the Viseur Home Page.[2]

DISCUSSION

Compression

The structure of the HTML pages, produced with the methods proposed here, is highly redundant. This may be important, in particular for three reasons.

1. Users managing a databank that stores alignment data and serves HTML representations may want to compress that data, in order to reduce the storage capacity needed, then to decompress the data on request, as sent.
2. A few users may access data through a modem. This hardware usually proposes compression of the data on the fly.
3. In future, network technology (including hardware and software) may evolve to propose compression of transmitted data, to make a higher profit on the actual bandwith. We agree that this consideration is purely hypothetical, but we note that, if this evolution occurs, our representations will take advantage of it, transparently.

Because there are only about 20 residues, plus one code for gaps, choosing one, or n bytes to encode a position in an alignment, will not increase the compressed size of the representation n times. Instead, according to the way compression algorithms work (Ref. 10 reviews data compression algorithms), the size will increase by approximately $21 \times n$ bytes.

To illustrate this, we used the Ziv–Lempel algorithm,[11] as implemented in gzip (gzip is a compressor available under the terms of the GNU software license) and obtained a 93–96% compression. This is typical of our data, and much higher than what is observed, for example, for English natural text (60–70%). Table 1 presents the amount of data needed for each representation.

Should server-side compression of HTML pages be supported by every browser, we would be able to send compressed files through the network, and let the browser decompress the file before it displays the content. Currently, this works solely with Netscape browsers under Unix. To illustrate this principle, an example is available as a compressed HTML alignment page, to be viewed on Unix systems only (compressed size, 6 979 bytes; average, 1.06 bytes per position). For this to work, the WWW server should encode files ending with `.gz` as MIME `x-gzip` documents. Transfer time through the network becomes negligible versus decompression and rendering time (browser and machine dependent) and it is not clear whether there is a great benefit to use it in this case, except to reduce the bandwidth consumed between the server and the client.

Advantages and drawbacks

We summarize, in Table 2, advantages and drawbacks of the methods we have described in this article.

As suggested by Table 2, these two methods are partially orthogonal, with respect to the public they target. Graphical alignments are viewed best with HTML2 browsers, which still constitute, until each browser of this type is updated to a more recent release, an important part of the audience. Style sheet alignments are targeting that part of the public that uses more recent browsers. Researchers who would like to publish MSAs on the WWW should propose both alternatives to visitors to their sites. Moreover, they should clearly describe what page the user should access, according to his or her browser.

CONCLUSION

In this article, we have described two novel methods for the visualization of multiple sequence alignments (MSAs) using HTML. These methods construct colored alignments, possibly hyperlinked. They differ depending on the audience they can

Table 1. Amount of data needed by each representation, per residue, in an MSA

	Text (MSF)	Colored text	Simple image	Colored graphics	Style sheets
Not compressed (bytes/position)	1	26 or >31[a]	118	≥20[b]	≥24
gzip (bytes/position)	1.06	2.03	118	1.06	1.18

[a] A total of 31 bytes is needed to represent colors for which no symbolic name is defined. This number of bytes is required for expression such as <FONT COLOR="501000">A</FONT>, that represent one colored position.
[b] A total of 20 bytes for links to images is a minimum. Hyperlink inclusion increases this minimal value. Twenty bytes is obtained for <IMG SRC="i/_.gif">.

Table 2. Comparison of graphical and CCS1 MSA representations

	Graphical alignment	CSS1 alignments
Can be viewed with graphical HTML2.0 browsers (Netscape 3-, Internet Explorer 2-, etc.)	Yes	No
Can be viewed with minimally complaint CSS1 browsers	Yes (may be slow)	Yes (display should be efficient)
Colors per residue class	Yes	Yes
Specific color on some residues	Yes	Yes
Hyperlink support	Yes	Yes
Special graphical symbols	Yes	No

reach. The first method uses HTML2.0 language. This ensures the production of HTML pages that will be rendered the same way by every HTML2.0 graphical browser. The second method we propose relies heavily on cascade style sheets, as specified in the W3C CSS-1 recommendation. Thus, it is intended to be used with recent and future browsers. At the present time, according to the type of browsers available to users, both methods are complementary. We expect this situation to evolve gradually, until such a time that the style sheet method can be used alone. Finally, we have implemented these novel methods, restrained to protein sequences, in the Viseur program and have provided a WWW conversion service, from protein MSAs (MSF file format) to HTML (colored text, graphic, style sheet). The service is available from the URL http://www.lctn.u-nancy.fr/viseur/services.html.

ACKNOWLEDGMENTS

Fabien Campagne acknowledges the supercomputing center Centre Charles Hermite for his Ph.D. grant, and Christophe Chipot for proofreading this manuscript.

REFERENCES

1 Rodriguez-Tome, P. *The BioCatalog Alignment Editing and Display*. European Bioinformatic Institute, WWW: http://www.ebi.ac.uk/biocat/biocat_ebi.html, October 1993–1997; includes references to publications

2 Campagne, F., Bernassau, J.-M., and Maigret, B. *The Viseur Program.* Laboratoire de Chimie théorique de Nancy, WWW:http://www.lctn.u−nancy.fr/viseur/viseur.html, 1995–1997, Viseur project home page

3 Horn, F., Weare, J., Beukers, M.W., Hörsch, S., Bairoch, A., Chen, W., Edvardsen, Ø., Campagne, F., and Vriend, G. GPCRDB: An information system for G protein-coupled receptors. *Nucleic Acids Res.* 1998, **1,** 275–279

4 Numerous authors. *HTTP Specifications and Drafts.* W3C, WWW:http://www.w3.org/Protocols/Specs.html. November 1997

5 di Tommaso, M. *The Alignment Colour Viewer.* European Bioinformatic Institute, WWW:http://www.ebi.ac.uk/htbin/visalign.pl. 1995

6 Barton, G.J. ALSCRIPT—A tool to format multiple sequence alignments. *Protein Eng.* 1993, **6,** 37–40. WWW: http://barton.ebi.ac.uk/barton/servers/amas_server.html

7 Attwood, T.K., Payne, A.W.R., Michie, A.D., and Parry-Smith, D.J. *A Colour INteractive Editor for Multiple Alignments—CINEMA. EMBnet.news* **3,** WWW:http://ben.vub.ac.be/embnet.news/vol3_3/software.html. 1997

8 Attwood, T.K., Payne, A.W.R., Michie, A.D., and Parry-Smith, D.J. *A Colour INteractive Editor for Multiple Alignments—CINEMA.* UCL's Bioinformatics server, WWW:http://www.biochem.ucl.ac.uk/bsm/dbbrowser/CINEMA2.02/. November 1997

9 Håkon Wium, L. and Bos, B. *Cascading Style Sheets Page*. W3C, WWW:http://www.w3.org/Style/css/. 1995–1997

10 Lelewer, D.A. and Hirschberg, D.S. *Data Compression.* Information and Computer Science, University of California, Irvine, WWW:http://www.ics.uci.edu/dan/pubs/DataCompression.html. 1997

11 Ziv, J. and Lempel, A. A universal algorithm for sequential data compression. In: *IEEE Trans. Inform. Theory*, 1977, pp. 337–343