BIOINFORMATICS APPLICATIONS NOTE Vol. 21 no. 4 2005, pages 537–539
doi:10.1093/bioinformatics/bti054

Circular genome visualization and exploration using CGView

*Paul Stothard and David S. Wishart**

Department of Biological Sciences and Computing Science, University of Alberta, Edmonton, Alberta, Canada T6G 2E8

Received on June 7, 2004; revised on August 31, 2004; accepted on September 16, 2004
Advance Access publication October 12, 2004

ABSTRACT

Summary: CGView (Circular Genome Viewer) is a Java application and library for generating high-quality, zoomable maps of circular genomes. It converts XML or tab-delimited input into a graphical map (PNG, JPG or Scalable Vector Graphics format), complete with sequence features, labels, legends and footnotes. In addition to the default full view map, the program can generate a series of hyperlinked maps showing expanded views. The linked maps can be explored using any Web browser, allowing rapid genome browsing and facilitating data sharing.

Availability: CGView (the standalone application, library or applet), sample input, sample maps and documentation can be obtained from http://wishart.biology.ualberta.ca/cgview/

Contact: david.wishart@ualberta.ca

Graphical maps of DNA sequence features provide a means to quickly view the characteristics of specific genomic regions or genes. The annotations produced by several different analyses are often displayed on these maps simultaneously, and the resulting feature juxtaposition can be very useful. For example, gene prediction results for a particular region might be deemed to be more reliable if it is apparent that the proposed gene segments also show elevated sequence conservation across species. Maps also allow gene context to be visualized, and thus can be used to identify candidate operons in bacterial species. Several software packages have been described for producing graphical sequence maps. The Microbial Genome Viewer (Kerkhoven *et al*., 2004) is a Web-based tool for generating maps in Scalable Vector Graphics (SVG) format. Users can construct maps using annotations provided by the program, or they can upload custom annotations. GenomePlot (Gibson and Smith, 2003) is a standalone program that generates GIF, TIFF, JPG and PostScript maps from a simple tab-delimited feature file. Finally, GenoMap (Sato and Ehira, 2003) is a standalone program for generating PostScript maps of microarray expression and gene position data. Unlike the Microbial Genome Viewer and GenomePlot, GenoMap is specifically designed for circular genomes. One limitation of these existing software packages is that a fairly restricted set of options is provided for controlling the appearance and organization of their genomic maps. For example, GenomePlot and GenomeMap do not allow the color of individual features to be specified in the input file. In many cases, other options for controlling map appearance can be set using a GUI or Web interface. However, these options cannot easily be manipulated using another program, making it difficult to incorporate the mapping software into other analysis programs or pipelines. None of these existing programs supports circular map feature labeling or hyperlinking. Similarly, none supports circular map zooming in a way that allows very closely spaced features to be fully resolved.

Here we describe CGView (Circular Genome Viewer), a Java application that can be used to generate both static and interactive graphical maps of circular DNA molecules, such as plasmids and bacterial genomes. The output of CGView is highly customizable. Feature labels can be displayed and hyperlinked, and regions of interest can be zoomed to almost any size. In addition to serving as a standalone application, CGView can be incorporated as a Java library into other programs, using the provided Application Programming Interface (API). A Java applet that uses the CGView library to display graphical maps of bacterial genomes can be accessed via the CGView homepage.

CGView can generate maps from three different types of textual input. For complete control over all aspects of map appearance, input can be supplied in the form of an Extensible Markup Language (XML) file. The various elements and attributes in the XML file are used to describe sequence features (position, type, name, color, label font and opacity). Additional optional XML attributes can be included, to control global map characteristics, and to add legends and a title. Alternatively, CGView can read feature information from tab-delimited text files. Two tab-delimited formats are supported: the CGView format and the NCBI protein table format. The CGView format allows up to 11 values to be supplied for each feature, to specify the location and appearance of the feature, and to provide hyperlink and mouseover information.

*To whom correspondence should be addressed.

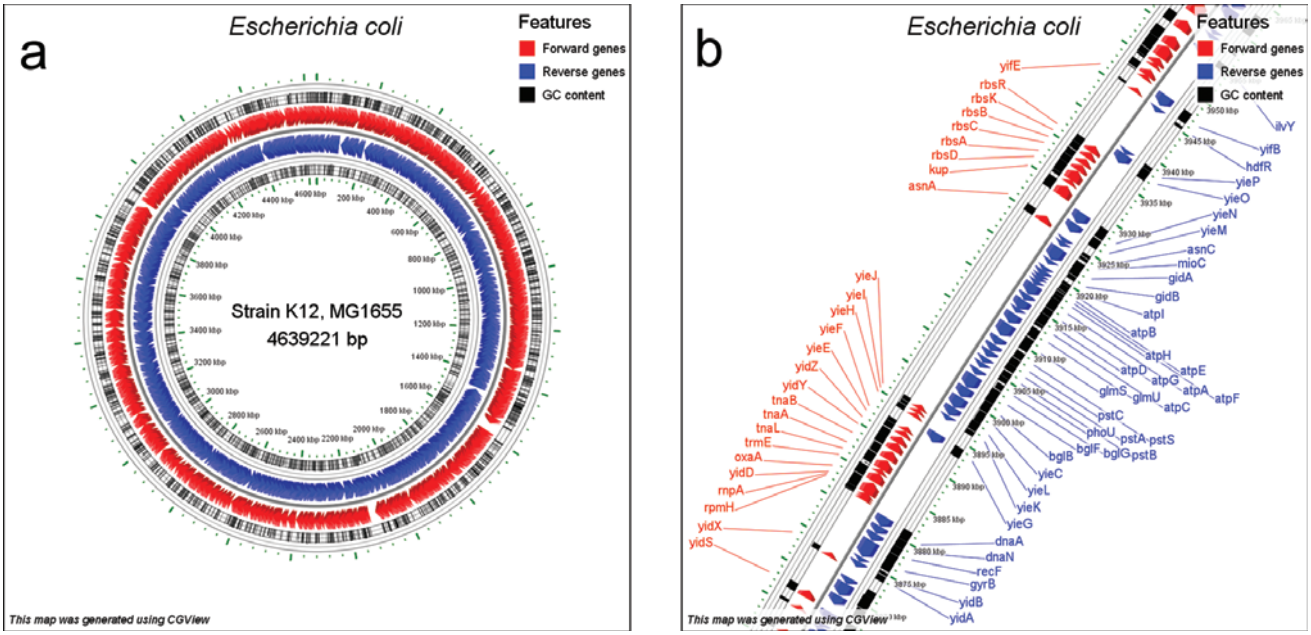

Fig. 1. (**a**) A CGView map of the *Escherichia coli* genome, with genes marked as arrows. The %GC content is shown for each gene in an adjacent feature track. (**b**) The same map rendered at an expanded size and centered on a region of interest.

The NCBI protein table format consists of eight fields of information for each gene, including location, name, unique identifier, COG functional category (Tatusov *et al*., 2000) and function. Protein table files, which typically end in a '.ptt' extension, are available for all completed bacterial genomes from ftp://ftp.ncbi.nih.gov/genomes/Bacteria/. Several command line options can be used in conjunction with the tab-delimited formats, to make adjustments to the appearance of the map. More detailed descriptions of the input formats and a summary of the command line options are available on the CGView website.

By default, CGView returns completed maps as PNG images. JPG or SVG files can be created instead, using the 'format' command line option. Maps can depict an entire circular genome, or a portion of a genome at an expanded size. The 'zoom' and 'center' command line options determine which type of map is drawn. By default, the zoom value is set to one, and the entire map is rendered. When a zoom value greater than one is supplied, the radius of the sequence backbone circle is multiplied by the zoom value, and the canvas is positioned over the center base (or base one by default). As shown in Figure 1a and 1b, the zooming capability can be used to greatly increase the detail of specific map regions.

CGView can produce a single image, as described above, or an image series consisting of a single complete view of the map, and several expanded views. Each of the images is created with an accompanying HTML file, which is used to link each tick mark in the map to other images in the series. This linking allows a region of interest to be viewed at an expanded size, simply by clicking the nearest tick mark. The HTML file also allows feature labels to be linked to the relevant websites and mouseover information. Website URLs and mouseover text can be specified for each feature in the XML or tab-delimited input supplied to CGView. When an NCBI protein table file is used as an input, each feature label in the resulting image series is automatically hyperlinked to the appropriate GenBank record. The completed image series can be explored by loading the resulting HTML files into a Web browser, and the maps can be made available to others simply by placing them in a publicly accessible HTML directory. When a series of linked maps is created, a combination of PNG and SVG output is generated. The PNG images serve as the default view, because the PNG format provides good compression without image degradation, and PNG images are compatible with current Web browsers. The SVG versions of the maps can be accessed by clicking the 'View SVG Map' button, which is located at the bottom of each map in the series. Most Web browsers can display SVG content using the SVG plugin available from Adobe (http://www.adobe.com). Linked maps created using CGView have recently been incorporated into the CyberCell database, a repository of information relating to *Escherichia coli* (Sundararaj *et al*., 2004).

To facilitate the integration of CGView into other sequence analysis or presentation programs written in Java, a complete API is provided. All of the map characteristics that can be specified using XML and command line options can

also be described using the API. The PlasMapper Web server, which automatically annotates plasmid sequences, uses this API to generate the maps it provides (Dong *et al*., 2004). Documentation for the CGView API is available in Javadoc format on the CGView website. A Java applet built using the API is also available, as a further example of how other Java programs can make use of the CGView library. Non-Java programs can also make use of CGView, by issuing commands through its command line interface.

Subsequent versions of CGView will accept GenBank and EMBL sequence records as input. New options will also be added for controlling how the feature types in these records are drawn.

ACKNOWLEDGEMENT

Funding for this project was provided by Genome Prairie (a division of Genome Canada).

REFERENCES

Dong,X., Stothard,P., Forsythe,I.J. and Wishart,D.S. (2004) PlasMapper: a web server for drawing and auto-annotating plasmid maps. *Nucleic Acids Res*., **32**, W660–W664.

Gibson,R. and Smith,D.R. (2003) Genome visualization made fast and simple. *Bioinformatics*, **19**, 1449–1450.

Kerkhoven,R., van Enckevort,F.H.J., Boekhorst,J., Molenaar,D. and Siezen,R.J. (2004) Visualization for genomics: the Microbial Genome Viewer. *Bioinformatics*, **20**, 1812–1814.

Sato,N. and Ehira,S. (2003) GenoMap, a circular genome data viewer. *Bioinformatics*, **19**, 1583–1584.

Sundararaj,S., Guo,A., Habibi Nazhad,B., Rouani,M., Stothard,P., Ellison,M. and Wishart,D.S. (2004) The CyberCell Database (CCDB): a comprehensive, self-updating relational database to coordinate and facilitate *in silico* modeling of *Escherichia coli*. *Nucleic Acids Res*., **32**, D293–D295.

Tatusov,R.L., Galperin,M.Y., Natale,D.A. and Koonin,E.V. (2000) The COG database: a tool for genome-scale analysis of protein functions and evolution. *Nucleic Acids Res*., **28**, 33–36.