BIOINFORMATICS APPLICATIONS NOTE Vol. 25 no. 10 2009, pages 1333–1334
doi:10.1093/bioinformatics/btp153

Sequence analysis

DASher: a stand-alone protein sequence client for DAS, the Distributed Annotation System

David N. Messina and Erik L. L. Sonnhammer*

Stockholm Bioinformatics Centre, Stockholm University, 10691 Stockholm, Sweden

Received on November 10, 2008; revised on February 26, 2009; accepted on March 14, 2009

Advance Access publication March 17, 2009

Associate Editor: Dmitrij Frishman

ABSTRACT

Summary: The rise in biological sequence data has led to a proliferation of separate, specialized databases. While there is great value in having many independent annotations, it is critical that there be a way to integrate them in one combined view. The Distributed Annotation System (DAS) was developed for that very purpose. There are currently no DAS clients that are open source, specialized for aggregating and comparing protein sequence annotation, and that can run as a self-contained application outside of a web browser. The speed, flexibility and extensibility that come with a stand-alone application motivated us to create DASher, an open-source Java DAS client. Given a UniProt sequence identifier, DASher automatically queries DAS-supporting servers worldwide for any information on that sequence and then displays the annotations in an interactive viewer for easy comparison. DASher is a fast, Java-based DAS client optimized for viewing protein sequence annotation and compliant with the latest DAS protocol specification 1.53E.

Availability: DASher is available for direct use and download at http://dasher.sbc.su.se including examples and source code under the GPLv3 licence. Java version 6 or higher is required.

Contact: erik.sonnhammer@sbc.su.se

Supplementary information: Supplementary data and all figures in color are available at *Bioinformatics* online.

1 INTRODUCTION

Biological data are accumulated and provided by a large number of laboratories across the world. As a result, today we have hundreds of different web sites with different interfaces and little integration among them. Researchers wanting to compare annotations from disparate sources, even those that relate to the same sequence, must aggregate those annotations themselves, and this is typically a manual, tedious and time-consuming process.

The Distributed Annotation System (DAS) was developed to overcome these problems by creating a standard protocol by which source databases could serve and client programs could access biological sequences and annotations (Dowell *et al*., 2001). By doing so, DAS establishes two huge advantages over the previous system: (i) DAS creates a standard way to access data, so all sources which comply with that standard can be viewed with the same software, and additional sources can be added effortlessly; (ii) DAS separates the databases responsible for serving data from the software which shows those data to the end user, so the software can be specialized for particular needs, and users can choose whichever way of looking at that data they feel is best.

Since its inception the DAS universe has grown broadly. There are different flavors of DAS suited to sequence, structure and protein interaction annotation. The DAS server registry allows users to view DAS-compliant data sources both manually through the web site and automatically through software (Prlić *et al*., 2007). Several DAS clients are available: SPICE (Prlić *et al*., 2005), CARGO (Cases *et al*., 2007), DASMI (http://dasmi.bioinf.mpi-inf.mpg.de/), Jalview (Clamp *et al*., 2004), PeppeR (http://biocomp.cnb.uam.es/das/PeppeR/), IGB (http://genoviz.sourceforge.net/), Pfam (Finn *et al*., 2008), MaDas (http://madas.bioinfo.cnio.es), Dasty2 (http://www.ebi.ac.uk/dasty) and Ensembl protview (Birney *et al*., 2006). However, only Dasty2, Ensembl and Pfam are focused on protein sequence annotation, yet run in a web browser rather than as a separate application, which can limit their flexibility.

Here we introduce DASher, a lightweight, stand-alone Java DAS client optimized for viewing protein sequence features. As a stand-alone application, DASher offers advantages in responsiveness and interactivity. For example, users can zoom in on a region, and zoom-dependent sequence rendering displays residues automatically at a sufficiently high zoom level. Also, users have full control to customize the appearance of the data being displayed.

2 OVERVIEW

2.1 Implementation

DASher is written in the object-oriented, platform-independent programming language Java (http://java.sun.com). It is based on the Sfixem platform (Chalk *et al*., 2004) and uses the Dasobert library to handle DAS stream input and output (http://www.spice-3d.org/dasobert/). DASher has been tested and runs on Apple OS X, Windows XP and GNU/Linux operating systems.

One-click installation is available via Java Web Start from the DASher website. DASher is licensed under the GNU General Public Licence.

2.2 User interface

To see annotations for a protein, the user enters a UniProt name or accession number in the search box. DASher automatically queries the DAS Registry for all active and validated servers which

*To whom correspondence should be addressed.

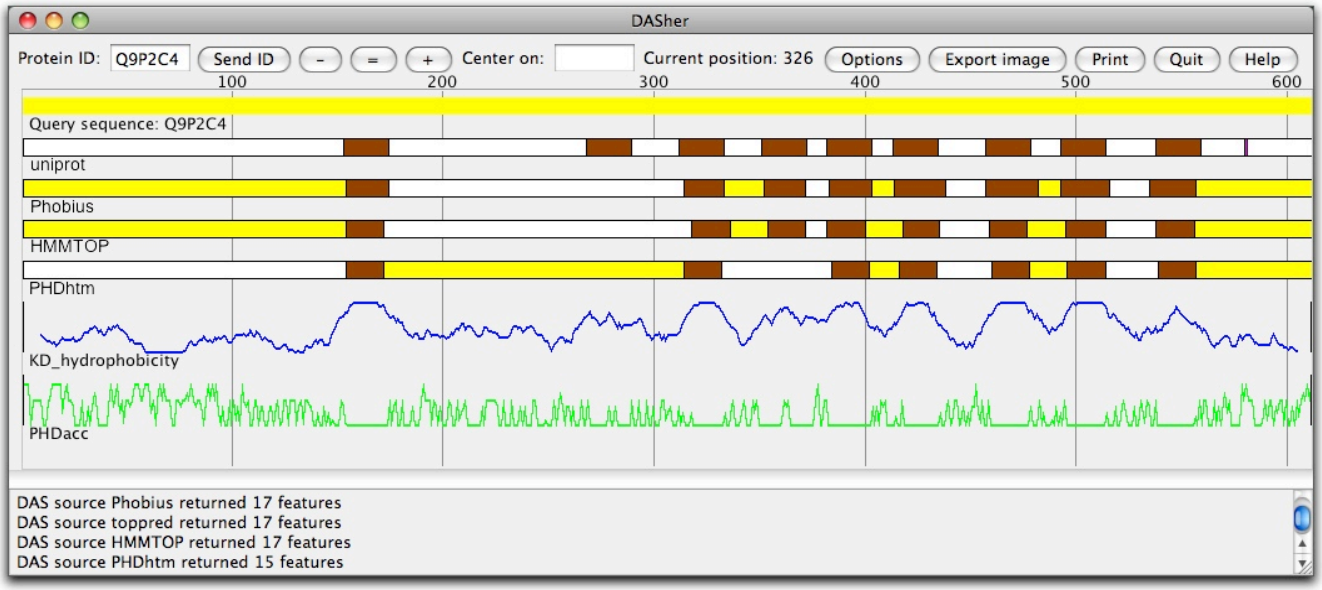

Fig. 1. The DASher main window. When a UniProt identifier is entered in the top left box, DASher fetches annotations from relevant DAS servers which are each displayed on a separate row, color-coded by feature type. In this example, we analyze the transmembrane topology of a largely unclassified human protein. Yellow denotes cytoplasmic regions; white, non-cytoplasmic regions; and brown, transmembrane regions. Tracks from top to bottom are the query sequence, UniProt, three transmembrane topology predictors, Kyte–Doolittle hydrophobicity (blue) and predicted solvent accessibility (green).

offer protein feature annotations, asks each of those servers for any information it may have on the requested sequence, and then displays those results in an integrated view in the main DASher window. For example, Figure 1 shows how one can examine in detail the transmembrane topology of a protein. In this case, three transmembrane domain prediction programs, a solvent accessibility predictor and a hydropathy plot all argue against the second transmembrane domain (residues 268–288) annotated in the UniProt record (track 2).

Each server's annotation appears as a separate track labeled with the name of the server. Depending on the nature of the data, a feature will be shown either as a box or as a line plot. Segments, such as protein domains, use the former representation, while annotations composed of continuous values such as hydrophobicity use the latter. All features of a given type are color-coded identically so that it is easy to identify when multiple servers have made the same annotation for a given region of a protein. There is an ongoing effort to standardize DAS servers around a sequence feature ontology (Jenkinson *et al*., 2008), and DASher uses this information to match annotations where possible.

Users can zoom in on a particular region of a sequence, either by clicking directly in a track to center around that point, by clicking or rubberbanding to select a region or by entering coordinates in the toolbar. A track information window pops up when pressing the middle mouse button. It provides details on the track source and the coordinates of each feature and allows feature colors to be changed. The display can be customized further via an options menu, in which tracks can be hidden or re-ordered, and the height of the track types can be set.

We have developed DASher to provide a lightweight, stand-alone DAS client application specialized for comparing protein sequence annotations quickly and easily. DASher is compliant with the latest DAS specification 1.53E (Jenkinson *et al*., 2008) and is freely available as open source.

ACKNOWLEDGEMENTS

The authors would like to thank Andreas Prlić and the Sonnhammer lab for valuable assistance and insightful discussion.

Conflict of Interest: none declared.

REFERENCES

Birney,E. *et al*. (2006) Ensembl 2006. *Nucleic Acids Res.*, **34**, D556–D561.

Cases,I. *et al*. (2007) CARGO: a web portal to integrate customized biological information. *Nucleic Acids Res.*, **35**, W16–W20.

Chalk,A.M. *et al*. (2004) Sfixem—graphical sequence feature display in Java. *Bioinformatics*, **20**, 2488–2490.

Clamp,M. *et al*. (2004) The Jalview Java alignment editor. *Bioinformatics*, **20**, 426–427.

Dowell,R.D. *et al*. (2001) The distributed annotation system. *BMC Bioinformatics*, **2**, 7.

Finn,R.D. *et al*. (2008) The Pfam protein families database. *Nucleic Acids Res*, **36**, D281–D288.

Jenkinson,A.M. et al. (2008) Integrating biological data—the distributed annotation system. *BMC Bioinformatics*, in press.

Prlić,A. *et al*. (2005) Adding some SPICE to DAS. *Bioinformatics*, **21** (Suppl. 2), ii40–ii41.

Prlić,A. *et al*. (2007) Integrating sequence and structural biology with DAS. *BMC Bioinformatics*, **8**, 333.