ABSTRACT

Summary: Programmatic access to the UniProt Knowledgebase (UniProtKB) is essential for many bioinformatics applications dealing with protein data. We have created a Java library named UniProtJAPI, which facilitates the integration of UniProt data into Java-based software applications. The library supports queries and similarity searches that return UniProtKB entries in the form of Java objects. These objects contain functional annotations or sequence information associated with a UniProt entry. Here, we briefly describe the UniProtJAPI and demonstrate its usage.

Availability: http://www.ebi.ac.uk/uniprot/remotingAPI
Contact: spatient@ebi.ac.uk

1 INTRODUCTION

The Universal Protein Resource (UniProt; The UniProt Consortium, 2008) provides a comprehensive and freely accessible central resource of protein sequences and functional annotation. UniProt data can be browsed online through http://www.uniprot.org. In addition, a number of services exist (Labarga et al., 2007) to retrieve the data in various formats including XML, RDF, fasta and flat file. In order to process the information contained in these formats, a parser needs to be written that transforms the input into suitable data structures. Sometimes this can be done without further knowledge of the UniProt entry structure; the protein sequence, for example, can simply be parsed from the corresponding sequence line. Difficulties arise if splice variants of a protein sequence are required, since both the comment and feature sections of the entry need to be parsed at the same time and a deeper understanding of the UniProt structures is needed. Furthermore, data format changes within UniProt can lead to significant maintenance overhead.

Therefore, a Java application programming interface (UniProtJAPI) has been developed to provide remote access for Java applications processing UniProt and related data, granting users access to all four major components in UniProt. These are the UniProt Knowledgebase (UniProtKB), the UniProt Reference Clusters (UniRef), the UniProt Archive (UniParc) and the UniProt Metagenomic and Environmental Sequences (UniMES) database. Additionally, the UniProtJAPI has been extended to take into account information referenced in UniProtKB entries, for instance InterPro (Mulder et al., 2007).

This is a resource of protein families, domains and sites, containing a number of member databases that derive protein signatures. The UniProtJAPI provides this InterPro-derived information such as scores, and start and end positions of the signatures, for each UniProtKB entry. The UniProtJAPI provides the ability to perform text and sequence similarity search across all this data, allowing users to access a single database entry with a given accession number, or whole entry sets matching defined criteria.

2 CENTRAL DATA STRUCTURES

The UniProtJAPI represents protein data from the UniProtKB as Java objects, which enables programmatic retrieval of functional annotation and sequence information. The object structures resemble the flat file and XML format structures to facilitate access for those users already familiar with the UniProtKB formats.

The five central data structures of the UniProtJAPI are: UniProtEntry, UniParcEntry, UniRefEntry, ProteinData and UniMesEntry. A UniProtEntry object represents a UniProtKB entry and provides methods to access all of its information. For example, getDescription().getProteinName() returns the protein name associated with this entry. A UniParcEntry object models a UniParc non-redundant sequence while a UniRefEntry object is a UniRef sequence cluster of 100, 90 or 50 percentage identity. The relationship between the former objects is represented in a ProteinData object. The ProteinData object associated with a UniProtKB entry is accessible via the getUniProtEntry() method. The UniParc entry and the three levels of UniRef sequence clusters relating to the UniProtKB entry are also accessible using the getUniParcEntry() and getUniRefEntry() methods. Additional information is available from the getInterProMatches() method that returns a list of InterProMatch objects. These represent the sequence patterns for the UniProtKB entry. Lastly, a UniMesEntry object provides access to a UniMES sequence and its related sequence patterns. A graphical representation of the UniProtJAPI object model along with the library documentation is available online.

3 LICENSE AND GETTING STARTED

UniProtJAPI is based on open-source technologies and the software is under the Apache License, version 2. UniProt data...
is under the Creative Commons Attribution-NoDerivs license.

To use the UniProtJAPI, a compressed data file (zip file) that contains all the Java classes has to be downloaded. The library requires a Java 5 runtime environment or above and an HTTP connection.

### 4 EXAMPLES

The UniProtJAPI may be used in a broad range of applications. It is used in the generation of IntAct (Kerrien et al., 2007) to retrieve a list of all proteins that have been updated between two dates. The IntEnz (Fleischmann et al., 2004) project uses it to retrieve all database cross-references for a specific EC number. Additional usages could include a user wanting to investigate characterized proteins that are similar to an unknown protein. In order to assist the user with this task, the similarity search tool Blast is integrated in the API.

#### Fig. 1. UniProtJAPI example source code showing the investigation of an uncharacterized UniMes sequence using Blast.

is under the Creative Commons Attribution-NoDerivs license.

To use the UniProtJAPI, a compressed data file (zip file) that contains all the Java classes has to be downloaded. The library requires a Java 5 runtime environment or above and an HTTP connection.

### ACKNOWLEDGEMENTS

We thank all Consortium members, in particular, N. Sklyar and E. Salazar.

**Funding:** UniProt is supported by the NIH grant 2 U01 HG02712-04, the EC FELICS grant (021902RII3), the NIH grant 1R01HG02273-01, by the Swiss Federal Government through the Federal Office of Education and Science and by the NIH grants and contracts HHSN266200400061C, NCICbBIG, and 1R01GM080646-01, and the NSF grant IIS-0430743.

**Conflict of Interest:** none declared.

### REFERENCES


