



EMBL

Client APIs

Examples

Bio::Das

Bio::DasLite

The registry

EMBL
Genome
Database

DAS client Perl APIs

Using DAS in Perl software development

Andreas Kähäri

`andreas.kahari@ebi.ac.uk`

`das@ebi.ac.uk`



EMBL

European Bioinformatics Institute
Wellcome Trust Genome Campus,
Hinxton, United Kingdom

26th April 2006



DAS client Perl APIs

Client APIs

Examples

Bio::Das
Bio::DasLite
The registry

A DAS client API is an aid to the software developer who is developing an application and that needs to make use of the DAS protocol.



DAS client Perl APIs (cont.)

Client APIs

Examples

Bio::Das
Bio::DasLite
The registry

The application might be a pure DAS viewer, such as Dasty or ProView, or it might be an application that uses DAS as one of its many data sources, such as the Ensembl genome browser.

In any case, being able to easily connect to a DAS server without having to deal with network programming and XML parsing will be appreciated by the programmer.



DAS client libraries (cont.)

There are a couple of useful tools available for client development in Perl:

Client APIs

Examples

Bio::Das
Bio::DasLite
The registry

Bio::Das: A client library by Lincoln Stein (CSHL) available from <http://biodas.org/> as well as from CPAN.

Bio::DasLite: An alternative library by Roger Pettett (Sanger), available from CPAN.

Registry service: The DAS registry at <http://das.sanger.ac.uk/registry/> has a SOAP interface that `SOAP::Lite` can talk to.



Example code fragments

Client APIs

Examples

Bio::Das

Bio::DasLite

The registry

The following slides show a short and very basic code fragment for each of these APIs.

For the two client libraries `Bio::Das` and `Bio::DasLite` we show how to connect to a DAS server and retrieve features.

For the registry SOAP service, we show how to get information about all registered DAS sources.



Bio::Das feature request

Use the `Bio::Das` API to query two separate DAS servers in parallel for the features on two segments on chromosome I of *C. elegans*:

```
use Bio::Das;

my $das = Bio::Das->new(5);

my @request = $das->features(
    -dsn => [
        'http://www.wormbase.org/db/das/elegans',
        'http://dev.wormbase.org/db/das/elegans',
    ],
    -segment => [ 'I:1,10000', 'I:10000,20000' ]
);
```

Client APIs

Examples

Bio::Das

Bio::DasLite

The registry

EMBL
Sanger
EBI



Bio::Das feature request (cont.)

Client APIs

Examples

Bio::Das

Bio::DasLite

The registry

The call to `features` returns a list of HTTP request objects. These has to be checked for success or failure. The package documentation contains example code that does this.



Bio::DasLite feature request

The (almost) equivalent code for the Bio::DasLite library:

```
use Bio::DasLite;

my $das = Bio::DasLite->new( { 'timeout' => 5 } );

my $das->dsn(
    [
        'http://www.wormbase.org/db/das/elegans',
        'http://dev.wormbase.org/db/das/elegans',
    ]
);

my $features =
    $das->features( [ 'I:1,10000', 'I:10000,20000' ] );
```

Client APIs

Examples

Bio::Das

Bio::DasLite

The registry

EMBL
Sanger
EMBL



Bio::DasLite feature request (cont.)

Client APIs

Examples

Bio::Das

Bio::DasLite

The registry

The `Bio::DasLite` library will soon also get the ability to communicate with the public DAS registry.

Until that code is committed, the way to talk to the registry is by means of `SOAP::Lite`.

EMSEMBL



Getting sources from the public registry

Client APIs

Examples

Bio::Das

Bio::DasLite

The registry

```
use SOAP::Lite;

my $sources =
  SOAP::Lite->service(
    'http://servlet.sanger.ac.uk/dasregistry/services/das'
    . ' :das_directory?wsdl' )->listServices();
```