

To Be Heard and Not Seen

Modelling the [DAP](#) and [DTS](#) services to fill a role similar to that of a [DNS](#), except for much more demanding operations over data, we often demonstrate Brown Dog via example client applications that utilize the services. In the posts below we show a number of examples of the DAP and DTS being used in general workflow systems such as [DataWolf](#), specialized workflow systems such as [PEcAn](#), content management/curation systems such as [Clowder](#), a command line interface, a web interface, a Chrome extension, a [bookmarklet](#), and from within HTML via a javascript library. In addition there is a [python library](#) with plans to also create interface libraries in R and Matlab. At its lowest level, however, the DAP and DTS are [RESTful](#) web services with an [API](#) that is directly callable from a wide variety of programming and scripting languages. For example the simple [bash script](#) below:

```
#!/bin/bash
url="http://dap-dev.ncsa.illinois.edu:8184/convert/$2/"

for input_file in `ls *.$1` ; do
    output_url=`curl -u user1:password -s -H "Accept:text/plain" -F "file=@$input_file" $url`
    output_file=${input_file%.*}.$2
    echo "Converting: $input_file to $output_file"

    while : ; do
        wget -q --user=user1 --password=password -O $output_file $output_url
        if [ ${?} -eq 0 ] ; then break ; fi
        sleep 1
    done
done
```

Our ultimate goal will be to include support for the DAP and DTS at the OS level and like the DNS, though largely invisible to the average user, be an essential part of the internet in terms of accessing, finding, and using data.