National Dog Database

TA Interface Testing Guide

Version 1.10

December 2025

Revision History

Date	Version	Description	Author
July 2005	1.0	First release	Equinox
August 2005	1.1	Updated with corrections from DIA	Equinox
September 2005	1.2	Updated to include Information on Unattended File Transfer in section 2.4 Updated to change document names Removed document date from reference table. All documents refer to the most current version.	Equinox
October 2005	1.3	P4. Correction from 3 to 2 types of file Page header and footer re-format and re-number. Addition to clarify the naming convention for a Full File (page 7 – 2.3, page 11 – 4.2)	Equinox
November 2005	1.4	Added section 2.4.4 describing how to use SCP to retrieve the XML To-Do List Report using SCP. Removed incorrect references to DSA	Equinox
December 2005	1.5	Inserted new section 2.4.2 to cover the use of PuTTY for creating private and public keys	Equinox
May 2006	1.6	Updated to reflect change of test server name to test.ndd.govt.nz and availability of additional Guidelines.	DIA
April 2010	1.7	Standardised SSH user ID examples. Removed out of date information.	Equinox
December 2011	1.8	DIA Logo Update	Equinox
July 2024	1.9	DIA Logo Update	Equinox
December 2025	1.10	DIA Logo Update	Equinox

Table of Contents

TA Interface Testing Guide

1. Introduction

The Department of Internal Affairs (DIA) National Dog Database (NDD) project requires the seventy plus Territorial Authorities (TAs) in New Zealand to provide information on dog registrations within their TA. The information is to be provided in XML format as described in the TA Interface Guidelines and the associated TA Interface Schema.

The NDD provides TAs and TA system suppliers with an interface testing site against which they can test their TA extract file. The objectives of this testing are:

- Test that the file is correctly formatted
- Test that the upload extract file updates the NDD database correctly
- Test the two types of extract file, incremental and full (reconcile and synchronize)
- Test the file transfer mechanism.

The following functionality is provided to enable the above:

- Batch upload
- Batch report
- To-do list
- NDD search.

1.1 Purpose

The purpose of this document is to describe the access and use of the interface testing site and provide some background information on the site.

1.2 Scope

This document only covers the accessing and use of the NDD interface testing website. See the references section below for other information pertaining to the NDD.

1.3 References

Reference	Author
Software Requirements Specification (SRS)	Equinox
Software Architecture Document	Equinox
NDD Project Glossary	Equinox
TA Interface Guidelines	Equinox
TA Interface Connection Guide	Equinox
TA Interface Schema	Equinox
State Services Commission Web Guidelines	SSC
State Services Commission e-GIF Standards	SSC

1.4 Overview

This document contains the following sections:

Section	Content
TA Extract File Transfer	Describes the steps involved in using a secure copy (scp) client application to transfer a TA extract file to the interface testing server. Also provides some background technical information.
Online Access	Describes the steps involved in using a web browser to access the online functionality of the NDD to view the results of an extract file upload.

1.5 NDD Interface Testing System

Equinox hosts the NDD production and interface testing environments.

Due to the reduced hardware resources used in the interface testing environment, it is expected that some elements of the system's performance will be slower than the NDD production system.

1.6 NDD Interface Testing System Availability

The NDD Interface server will generally be available on business days during normal business hours (9AM to 5PM). If there is a need to shutdown or restart the system during these times Equinox will endeavor to provide an email notification to the system users 30 minutes before the shutdown or restart.

1.7 NDD Interface Testing System UserIDs and Passwords

Each user of the test system will be supplied with a file transfer userid and password, and an online userid and password. These will be provided to each organisation separately.

Each organisation will also be given a TA and associated TA code to use as their test TA on the system.

There will also be other TAs used by Equinox and the DIA for their testing.

2. TA Extract File Transfer

2.1 Background

The interface testing site is using **S**ecure **Sh**ell (SSH) server to allow secure transfer of the TA XML Transfer files to the NDD. Two methods of user authentication are available, these being password authentication for interactive transfers, and public-key authentication for non-interactive usage.

It is anticipated the interactive (password) authentication would be primarily used during the interface testing phase to allow manual transfer of created test extract files.

The public-key authentication can be used for unattended file transfer (i.e. no user present to enter a password). The public-key authentication requires the TA to create a private/public key pair and to provide the public key to the NDD server.

Each TA has their own SSH userid and upon authentication will be placed into their own home directory (using the UNIX convention referred to as "~"). Each TA's home directory will also have a .ssh subdirectory (~/.ssh) which is used to stored the TA's public key to allow non-interactive file transfer (this is described in detail in section 2.4)

The extract file should follow the naming conventions described in the TA Interface Guidelines.

When the file is processed by the NDD it is moved out of the TA's home directory.

2.2 File Permissions

As SSH preserves file access permissions when a file is transferred, if the file does not allow other users to read the file the NDD application will be unable to read the file to process it. To ensure the file is readable the `chmod` command (in UNIX) can be used to allow the NDD Application to read it before it is sent. For example in UNIX the command:

```
chmod o+r extractfile.xml
```

makes the file `extractfile.xml` readable by the NDD application.

2.3 Interactive File Transfer Steps

The **S**ecure **C**o**p**y (scp) command can be used to transfer the extract file.

Interactive Command Line SCP Transfer

The basic format of a scp command to upload to remote host from your local machine is:

```
scp localfilespec myusername@remotehost:remotefilespec
```

For example the command:

```
scp extract.xml <ssh-user-id>@test.ndd.govt.nz:001-20050630-1.xml
```

Is used to copy the file `extract.xml` to TA 001's upload directory on the server `test.ndd.govt.nz` with the name `001-20050630-1.xml`. *Where* `<ssh-user-id>` *is the TA's SSH user name (as advised separately).*

(When transferring a Full file, the same process is used. The only difference is that the file name will contain the word "FULL" in the place of the batch sequence number (as well as the file type in the file header being FULL, not INCR.))

After the above is entered the password will be requested and if it is correct the transfer will occur. For example:

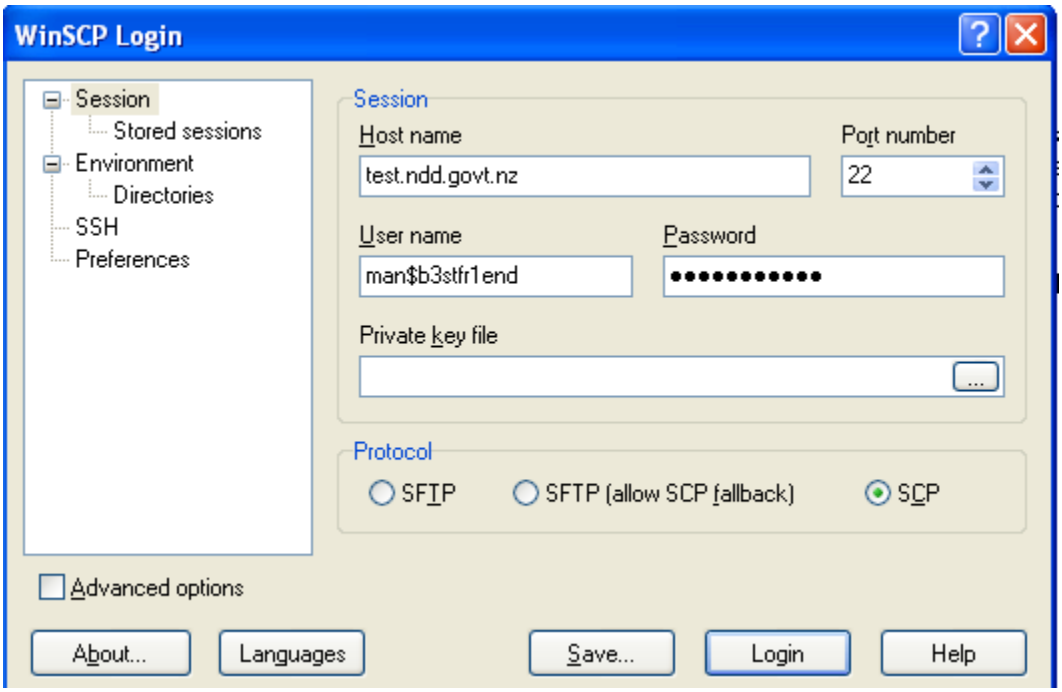
```
<ssh-user-id>@test.ndd.govt.nz's password:*********
extract.xml                          100% 5067     5.0KB/s   00:00
```

The scp uses SSH to securely transfer files between computers on a network. SCP and SSH are normally always included with Linux and Unix distributions. In a windows environment a command line SCP is available by installing the free Cygwin environment which provides Linux / Unix programs for the windows environment. (see http://www.cygwin.com/)

Interactive Graphical Interface SCP Transfer

Another option for the Windows environment is the WinSCP program. WinSCP is an open source SCP client for Windows. Its main function is the secure file transfer between a local and a remote computer. Beyond this, WinSCP offers basic file manager functionality. It uses Secure Shell (SSH) and supports the SCP protocol (see http://winscp.net/eng/index.php).

As shown in the figure below the WinSCP login dialog box is used to configure the session parameters, which can be saved and reconnected later.



The settings are:

Host name: test.ndd.govt.nz

Port number: 22

Username: The TA's SSH user name (as advised separately)

Password: The TA's SSH user password (as advised separately)

Protocol: SCP

2.4 Automatic (Unattended) File Transfers with Public-Key Authentication

The client applications identified above are useful during the interface testing phase, when the extra files will be manually placed onto the server. However, in the operational environment, the extract file uploads are expected to occur automatically after the end of the business day, an unattended SSH SCP command needs to occur.

The use of an unattended SCP command requires the use of public-key authentication to avoid the need for a user to enter a password.

To keep the instructions consistent the following definition of terms used in this section:

- *TA client system*: the system at the TA that is transferring the file to the NDD.
- *NDD server*: the NDD SSH Server.

Setting up the unattended transfer involves 3 major steps:

1. Creating the private and public keys
2. Add the public key to the TA's NDD server authorized_keys file
3. Automate the running of the SCP command at a predetermined time each business day

The details of each of these steps are described in the following sections.

2.4.1 Create Private and Public Keys

The actions shown are based on using the OpenSSH implementation of SSH. Key generation may vary under different implementations of SSH.

To confirm that OpenSSH is the SSH software installed on the TA client system. The ssh -V command should print a line beginning with OpenSSH, followed by other details.

```
$ ssh -V
OpenSSH_4.1p1, OpenSSL 0.9.7g 11 Apr 2005
```

A RSA key pair must be generated on the TA client system. The public portion of this key pair will be transferred to the NDD server, while the private portion needs to remain on a secure local area of the TA client system, by default in ~/.ssh/id_rsa, i.e. in the .ssh subdirectory under the home directory of the user the transfer will be performed under.

The key generation can be done with the OpenSSH ssh-keygen utility. The following commands create a .ssh directory under the root directory, make the directory non-public, and generate an RSA key file.

```
client$ mkdir ~/.ssh
client$ chmod 700 ~/.ssh
client$ ssh-keygen -f ~/.ssh/id_rsa -t rsa
Enter passphrase (empty for no passphrase): …
Enter same passphrase again: …
```

The meaning of the parameters passed to the ssh-keygen are:

- f <filename> filename of the key file, in this case "id_rsa"

- t <type> the type of key to create, in this case an "rsa" key

When the passphrase is requested do not enter anything[1], just press enter twice.

The file permissions should be locked down to prevent other processors or users from being able to read the key pair data. OpenSSH may also refuse to support public key authentication if the file permissions are too open.

```
$ chmod go-w ~/
$ chmod 700 ~/.ssh
$ chmod go-rwx ~/.ssh/*
```

2.4.2 *Creating the Private and Public keys using PuTTY and PuTTYgen*

A popular Windows FTP tool is PuTTY, which has an associated key generation utility called PuTTYgen which can be used to generate keys. However if the PuTTYgen tool is used there is an additional step required to get the key in the correct format for OpenSSH.

Keys can be in one of two formats, the SECCH format or the OpenSSH format. PuTTYgen uses the SECSH format for public keys, but its own unique format for private keys.

However PuTTYgen can read and write private and public keys for both OpenSSH and SECSH formats. To get the SECSH formatted key into OpenSSH format using PuTTYgen follow these steps:

1. Click the Load button and select the key file (you need to set file types to "All Files (*.*) for non-PuTTY keys to show up).
2. Use the Conversion menu and select "Export OpenSSH key".
3. To write an OpenSSH public key, the user needs to copy and paste the key shown at the top of the PuTTYgen Window into the id_rsa.pub file.

2.4.3 *Add the Public Key to the TA's NDD Server authorized_keys File*

The public portion of the RSA key pair must be copied to the NDD server that will be accessed by the TA client. The commands shown assume the public key information to be copied is located in the ~/.ssh/id_rsa.pub file on the client.

The commands will append the public key data into the ~/.ssh/authorized_keys file on the NDD server. Note that the .ssh directory has already been created on the NDD server under each TA's home directory.

Copy the public key to the TA's home directory on the NDD server using SCP

```
client$ scp ~/.ssh/id_rsa.pub <ssh-user-id>@test.ndd.govt.nz:
<ssh-user-id>@nddtest's password:*************
```

Where <ssh-user-id> *is the TA's SSH user name (as advised separately)*

Logon onto the NDD Server using SSH

```
ssh <ssh-user-id>@test.ndd.govt.nz
<ssh-user-id>@nddtest's password:*************
```

[1] If you wish to protect the private key with a passphrase, then use will need to be made of ssh-agent or similar utility to provide the passphrase when the automated SCP command is run. The configuration of this can be complex and will differ significantly for different Territorial Authority IT environments.

Append the public key to the authorized_keys file in the .ssh subdirectory

```
server$ cat id_rsa.pub >> .ssh/authorized_keys
```

By appending new public key data to the authorized_keys file multiple public keys may be used. Each public key entry must be on a different line in the authorized_keys file.

Set the permissions on the authorized_keys file

```
server$ chmod 600 .ssh/authorized_keys
```

Delete the copy of the public key from the TA's home directory, and exit the SSH session

```
server$ rm id_rsa.pub
server$ exit
```

Many different things can prevent public key authentication from working, so be sure to confirm that public key connections to the server work properly. Try logging onto the NDD server again using SSH. This time SSH should connect without any request for a password or passphrase.

```
client$ ssh <ssh-user-id>@test.ndd.govt.nz
server$
```

If there are issues at this stage using the "–v" can be passed to the ssh command to provide verbose output that may be useful in troubleshooting the issues.

SSH is strict about file and directory permissions and may refuse to use a private key file if the permissions to access that file are too lenient.

The SCP command can now be run without requiring the user to enter a password, for example:

```
scp extract.xml <ssh-user-id>@test.ndd.govt.nz:001-20050630-1.xml
extract.xml                               100% 5067     5.0KB/s   00:00
```

(When transferring a Full file, the same process is used. The only difference is that the file name will contain the word "FULL" in the place of the batch sequence number (as well as the file type in the file header being FULL, not INCR.))

2.4.4 Automate the Running of the SCP Command at a Predetermined Time

How the SCP command is automated depends on whether scheduling functionality is built into the vendor's dog control system, or the scheduling facilities of each TA's IT environment is used - such as the cron facility in UNIX / Linux environments, or the Task Scheduler in the Microsoft Windows environment.

The main consideration would be that the SCP command is called it has rights to access the private key file (`id_rsa` in our example) to allow the authentication to occur. The process that runs the SCP should be the owner of the private key file.

2.4.5 Retrieval of the XML To-Do List Report using SCP

If the TA has elected to receive the period processing To-Do List Report as XML, then the XML file will be placed in a subdirectory called `\out\` under the TA's home directory on the SSH Server. The TA also receives an email with a link to another copy of this XML file kept on the NDD Application Web Server.

The copy in the SSH `\out\` directory is overwritten each processing period, but the copies available on the NDD Application Web Server are available for an extended period of time (when they are removed is at the discretion of the NDD operations staff, and is only limited by disk storage).

The file on the SSH Server will always be called `ToDoListMessagesReport.xml`. The SCP command to retrieve the file is:

```
scp <ssh-user-id>@test.ndd.govt.nz:out/ToDoListMessagesReport.xml .
```

Note the space and full-stop at the end of the command – these are necessary.

3. Online Access

3.1 Background

The NDD online component is implemented as a web application which requires a web browser to access it. The web browser versions that have been tested by Equinox as at NDD "go live" date are:

- Microsoft Internet Explorer 5.5 Service Pack 2
- Microsoft Internet Explorer 6.0
- Mozilla FireFox 1.0

For security reasons the NDD online site uses the Secure Socket Layer (SSL) protocol, and when you first logon you will be presented with information about the digital certificate used on the server.

For the TA extract interface testing phase, the complete online functionality is being provided. The functionality required for testing includes:

- Logon
- Search NDD
- Request Report
- TA Extract Report
- To-Do List
- View Dog Details
- View Owner Details

3.2 Accessing NDD Online

The steps for accessing and using the NDD online application are described below. (Note the use of https).

In the browser enter https://test.ndd.govt.nz/NddWebApp/

As the NDD is using SSL, a dialog concerning the site's digital certificate will appear, similar to the one that follows. For the interface test server, a self signed certificate has been used; in the production system, a certificate signed by a Certification Authority is used.

Security Alert Screen

Press "Yes" to continue.

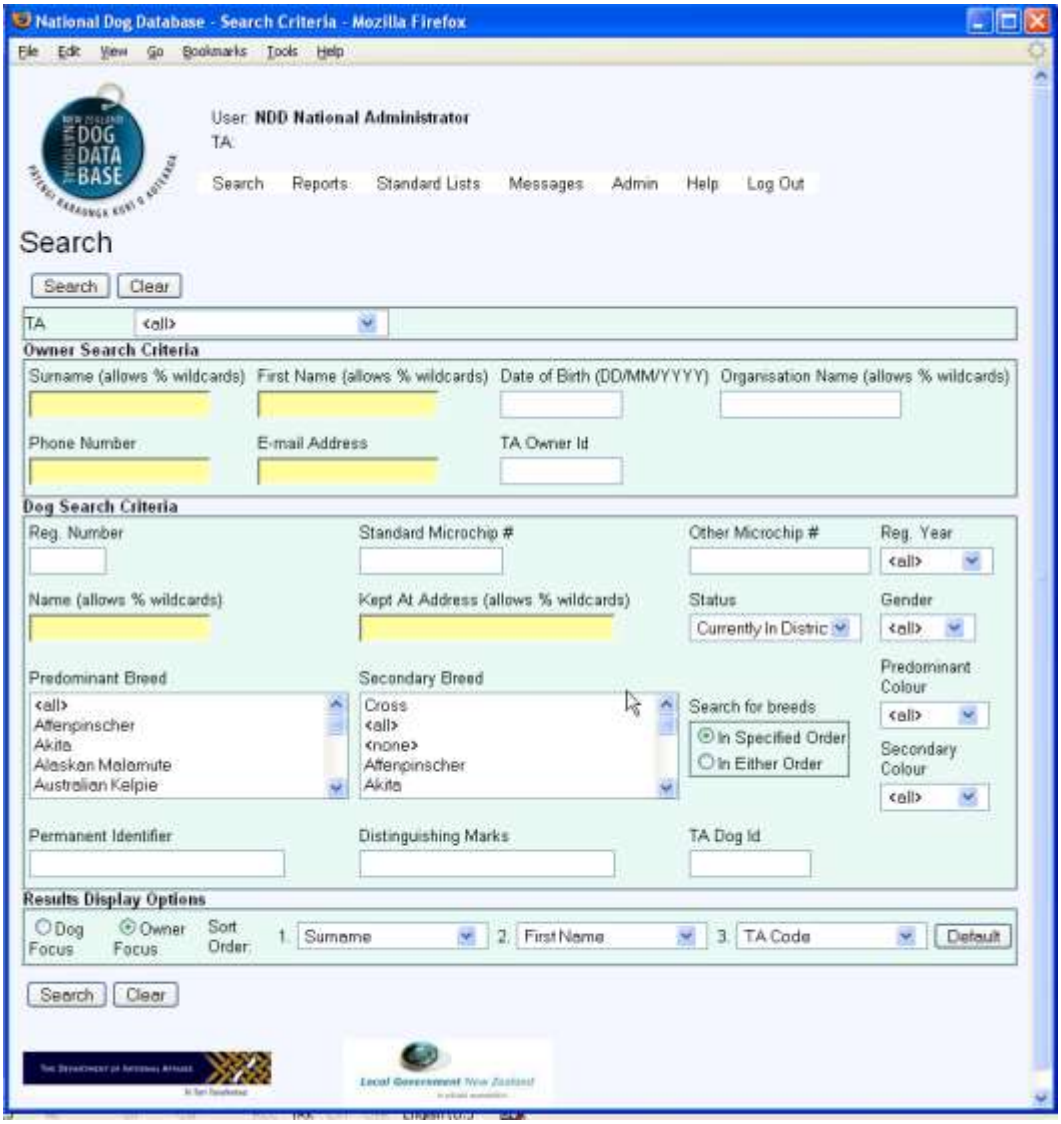
The NDD logon screen will then appear.

Log On Screen

Enter the online User name and password for your test TA (these will be provided separately). Note that if the browser session is inactive for 10 minutes the user is automatically logged out.

The NDD Menu will appear (*Search, Todo List, Reports, Log out, Help*) with the default **Search** screen displayed.

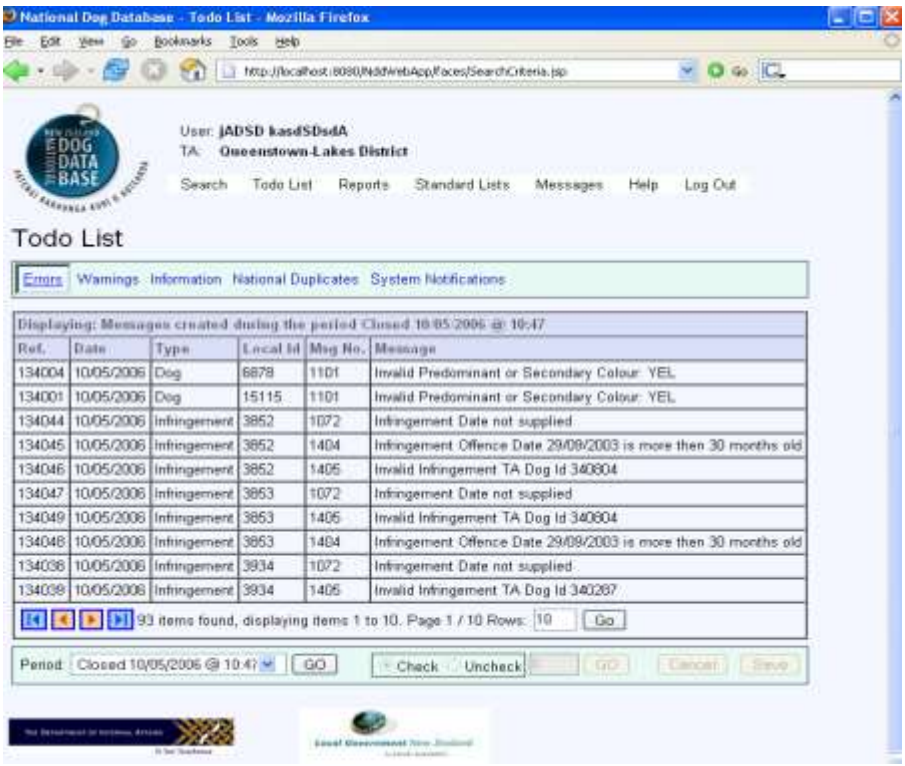
Search Screen



Todo List

The Todo List screen is used to present errors, warnings, or information messages associated with either the latest or all submitted TA Extract files for a TA.

Todo List Screen

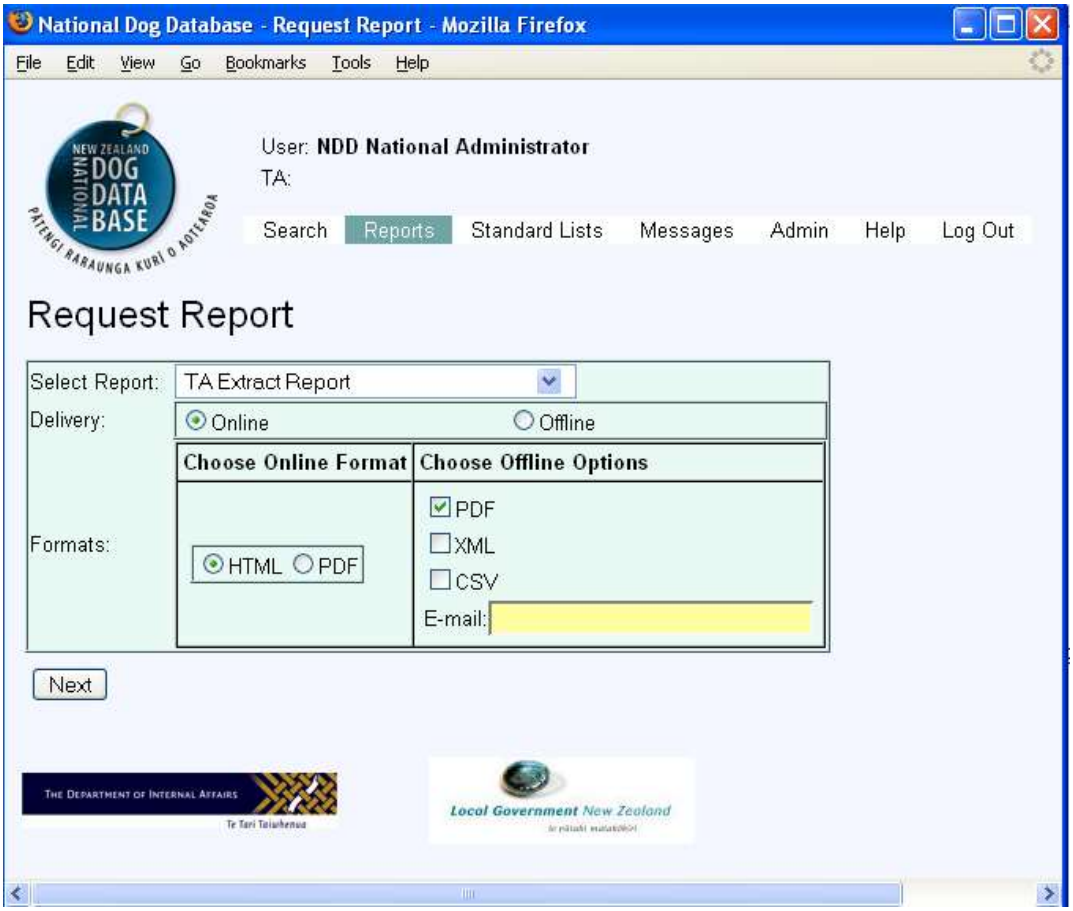


Reports

The Request Report screen allows the user to request a report that is delivered either online (in the browser) in HTML or PDF format, or offline (email advice of completion) in either PDF, XML or CSV format.

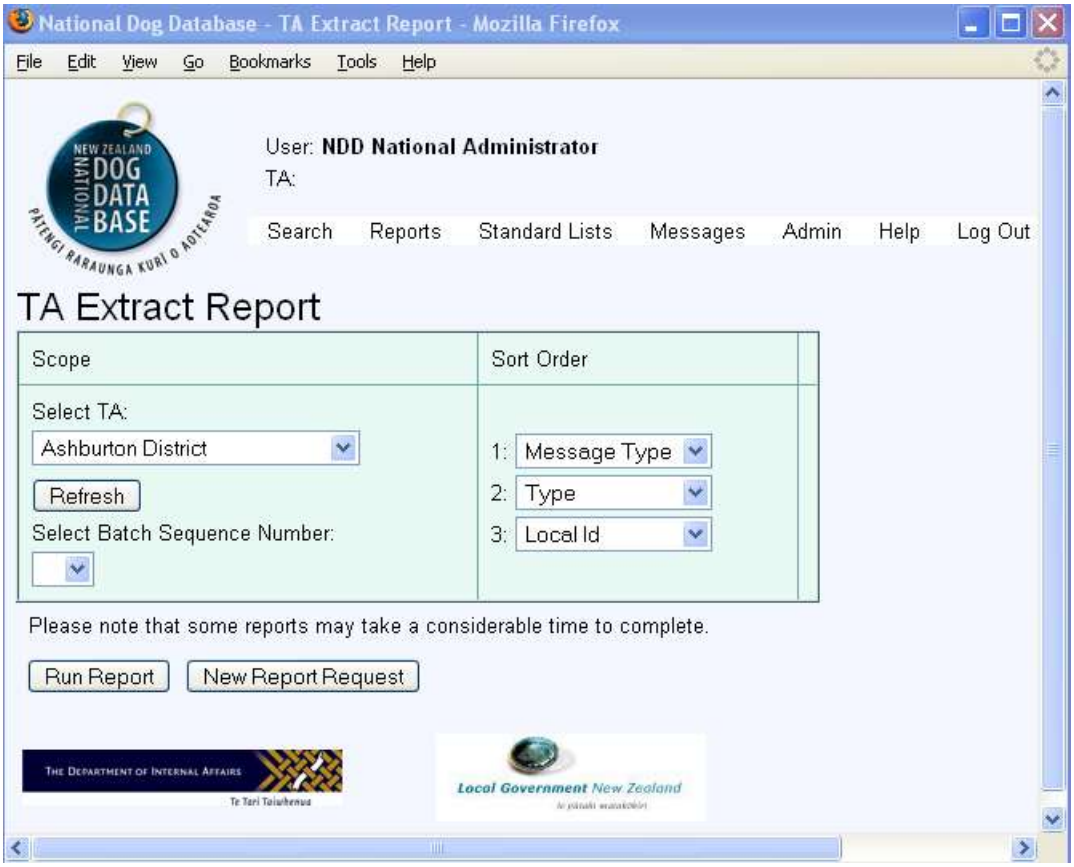
For the purposes of the Interface Testing, the ***TA Extract Report*** is available. This report shows the results of a batch upload. The following screen shots show the selection and display of an online HTML report. Select ***Online*** and ***HTML*** and press ***Next***.

Request Report Screen



Select TA from the list box (a TA user will only have their TA available), and press ***Refresh*** to populate the Batch Sequence Numbers for that TA. Select the ***Batch Sequence Number*** and ***Run Report***.

TA Selection Screen



When the report is created the following dialogue will appear. To view the report in a separate browser windows click on ***View Report***.

View Report Screen

The following is an example online HTML report.