



## **The DROID Application Programming Interface**

**Author:** Adrian Brown  
**Version:** 1  
**Date:** 6 September 2005

## Document Control

Author: Adrian Brown, Head of Digital Preservation

Document Reference: DROID-API-1

Issue: 1

Issue Date: 6 September 2005

## Document History

<b>Issue</b>	<b>Author</b>	<b>Date</b>	<b>Comments</b>
1	Adrian Brown	6 September 2005	Release version

## Contents

<b>1</b>	<b>INTRODUCTION .....</b>	<b>4</b>
<b>2</b>	<b>RUNNING DROID FROM THE COMMAND LINE.....</b>	<b>5</b>
<b>3</b>	<b>EXAMPLES .....</b>	<b>6</b>
3.1	Display a signature file version .....	6
3.2	Check signature file is up to date .....	6
3.3	Download a new signature file .....	6
3.4	Identify the formats of files in a list .....	6
3.5	Write output to CSV .....	6
3.6	Identify the formats of files listed in a File Collection file .....	7
<b>4</b>	<b>ERRORS .....</b>	<b>8</b>
4.1	Incorrect arguments .....	8
4.2	Web service failure .....	8

## 1 Introduction

This document describes the Application Programming Interface (API) for the DROID format identification tool. The full range of DROID functionality is exposed in a user-friendly manner via the Graphical User Interface. However, DROID also exposes its functionality via a command line API which is used:

- To allow DROID to be controlled from other software systems.
- To maximise throughput when performance is critical.

The DROID File Collection file and its XML schema, which is used by DROID to list files for identification and store the results, is described in Digital Preservation Technical Paper 1 (Automatic format identification using PRONOM and DROID).

## 2 Running DROID from the command line

To run DROID from a command line prompt:

1. Open up a command line prompt.
2. Navigate to the folder contain the DROID installation (e.g. C:/Program Files/DROID).
3. Launch DROID using `java -jar DROID.jar` followed by the required arguments (using no arguments launches the DROID GUI).

The available arguments for running DROID are listed below:

Action	Argument	Qualification
Display the command line options	-H	
Display the DROID application version	-V	
Display the signature file version	-V	The signature file name
Check whether the PRONOM website has a newer signature file	-C	The local signature file name
Download the latest signature file from the PRONOM website	-D	The filename to use for saving the downloaded signature file
Run the analysis on a comma separated list of files	-L	A comma-separated list of files to be identified
	-S	The signature file name
	-O	The filename for the results file. If no name is specified, the filename will default to <b>DROID out.xml</b> .
	-F	The format for the results file. The options are currently XML and CSV, with XML being the default.
Run the analysis on the files in a File Collection file	-A	The file name of the File Collection file
	-S	The signature file name
	-O	The filename for the results file. If no name is specified, the filename will default to <b>DROID out.xml</b> .
	-F	The format for the results file. The options are currently XML and CSV, with XML being the default.

## 3 Examples

### 3.1 Display a signature file version

To check that the signature file `SigFile_v3.xml` is well-formed and to display its version number:

```
java -jar DROID.jar -VSigFile_v3.xml
```

### 3.2 Check signature file is up to date

To check the version number of signature file `SigFile_v3.xml` and to compare it with the latest one available on the PRONOM web service:

```
java -jar DROID.jar -CSigFile_v3.xml
```

### 3.3 Download a new signature file

To download a new signature file and to save it as `SigFile_v4.xml`:

```
java -jar DROID.jar -DSigFile_v4.xml
```

### 3.4 Identify the formats of files in a list

To identify the formats of the files `C:\temp\file1.doc`, `C:\temp\file2.xls`, `C:\temp\file3.tif`:

```
java -jar DROID.jar -  
LC:\temp\file1.doc,C:\temp\file2.xls,C:\temp\file3.tif -  
SSigFile_v3.xml
```

This will generate a file called `DROID_out.xml` containing the results in the DROID File Collection file XML format, which can be displayed by DROID running in GUI mode.

### 3.5 Write output to CSV

The above example is repeated below, this time the output is written to the file `DROID_result.csv` in CSV format:

```
java -jar DROID.jar -  
LC:\temp\file1.doc,C:\temp\file2.xls,C:\temp\file3.tif -  
SSigFile_v3.xml -ODROID_result -FCSV
```

### 3.6 Identify the formats of files listed in a File Collection file

Listing files on the command line becomes very cumbersome when these become numerous. It may be preferable to create a list of files and record this in a DROID File Collection file. For example, the file `list.xml` would contain:

```
<FileCollection>
  <IdentificationFile Name="C:\temp\file1.doc" />
  <IdentificationFile Name=" C:\temp\file2.xls" />
  <IdentificationFile Name=" C:\temp\file3.tif" />
</FileCollection>
```

A DROID File Collection file can be created and viewed using the DROID GUI.

To run the file format identification on these files type:

```
java -jar DROID.jar -AC:\temp\list.xml -SSigFile_v3.xml
```

This would give exactly the same output as example 3.4 above.

## 4 Errors

### 4.1 Incorrect arguments

The following error is displayed when the expected command line arguments are not provided

```
Fatal Error: The command line arguments were incorrectly  
formed
```

Make sure that:

- Blank spaces are **only** included between command line arguments. In particular, file names with blank spaces in them cannot be used as part of the command line. To run DROID identification on a file that contains blank spaces in the name, include it in a file list file and then use the **-A** option.
- All arguments are uppercase as shown in the table above.
- Any file names used in the qualification of command line arguments exist.

### 4.2 Web service failure

When either the **-C** or the **-D** arguments are used, DROID will use the PRONOM web service. If this is not working, then one of the following errors will be displayed:

```
WARNING: Unable to get signature file version ...  
WARNING: Unable to download signature file ...
```

If this occurs, make sure that:

- The PRONOM web service is available.
- The URL for the PRONOM web service is correctly specified in the DROID configuration file (DROID\_config.xml), i.e. that the **<SigFileURL>** element contains the correct URL.