Signature Library - Introduction

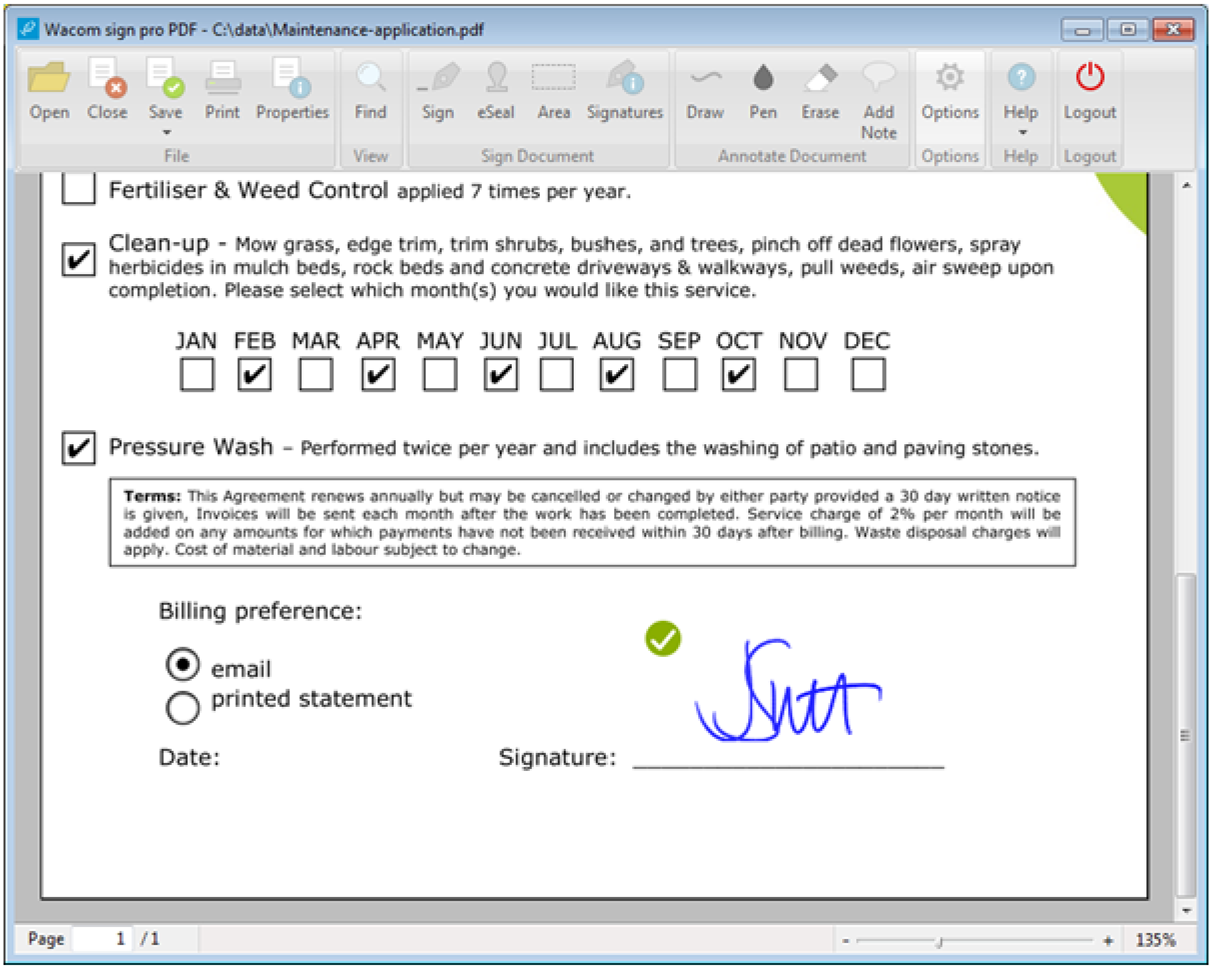
Contents

Introduction

Wacom's Signature SDK includes the Signature Library used to capture handwritten signatures from a pen tablet. The Signature Library simplifies the interaction with Wacom pen tablets and provides API to manage and display signatures. While this introduction is based on the Windows version of the Signature Library, the principles are also applicable to the mobile versions.

The Signature Library is used by the Wacom sign pro PDF application:

:



☐ **Fertiliser & Weed Control** applied 7 times per year.

☑ **Clean-up** - Mow grass, edge trim, trim shrubs, bushes, and trees, pinch off dead flowers, spray herbicides in mulch beds, rock beds and concrete driveways & walkways, pull weeds, air sweep upon completion. Please select which month(s) you would like this service.

JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEP	OCT	NOV	DEC
☐	☑	☐	☑	☐	☑	☐	☑	☐	☑	☐	☐

☑ **Pressure Wash** – Performed twice per year and includes the washing of patio and paving stones.

Terms: This Agreement renews annually but may be cancelled or changed by either party provided a 30 day written notice is given, Invoices will be sent each month after the work has been completed. Service charge of 2% per month will be added on any amounts for which payments have not been received within 30 days after billing. Waste disposal charges will apply. Cost of material and labour subject to change.

Billing preference:

● email
○ printed statement

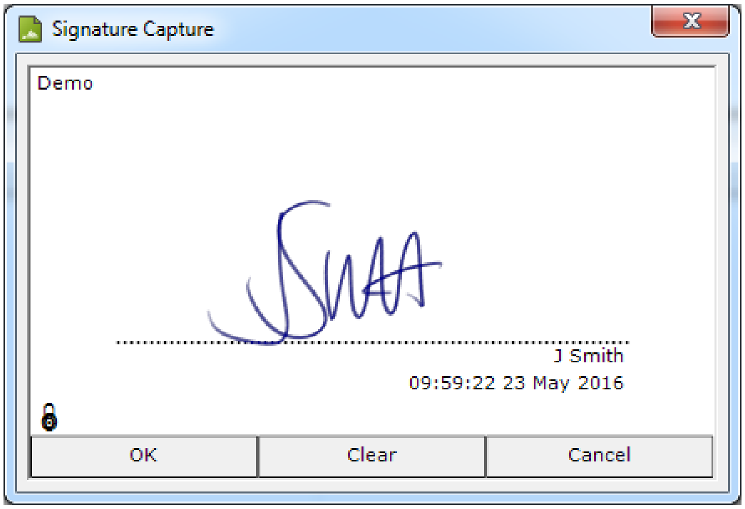
Date:

Signature:

Sign pro PDF uses the Signature Library API to capture and render signatures. To avoid a common misconception, note that the Signature Library does not include PDF related API and sign pro PDF uses a separate SDK to display and modify PDF documents.

Signature Capture

An application using the Signature Library will provide a way of starting signature capture, for example by displaying a ‘sign’ button. The resulting call to the signature capture function will display a dialog:



The signature capture function isolates the application from the type of pen input device. The Signature Library detects the type of connected device and automatically runs the code needed to communicate with it. Usually the pen input device will be an STU tablet such as the STU-540 or a Wintab device such as the DTU-1141. From an interface perspective these are very different devices but the Signature Library allows an application to use them in a common way.

The dialog shows the username, reason and timestamp which will be saved with the signature. While signing, the user can see the pen trace and clear it to sign again if needed before accepting the signature with the OK button.

A dialog is used so that the signing process can be independent of the underlying document to which the signature will be applied. For example, if the document view has been zoomed the signature can still be captured in the normal way and then adjusted to fit into the document display area.

To provide a high quality pen and ink experience the signature components use Wacom’s Ink Layer Language (WILL) technology for signature capture and image creation.

The Signature Library installation includes common language translations. The Windows installation language is detected and the relevant language is selected automatically.

Signature Object

The Signature Library creates a Signature Object to hold a captured signature. A signature can be handled by an application in its native binary format (Forensic Signature Stream - FSS) or in the Base64 text encoded format such as the following example:

Base74 SigObj

```
RlOArRoBGwECGEVDFxYZVAUHCQoDBggEDQxPFB0cGhsYFQIiBSAl75yzZny19rFciTS2f1ymdEac
0QlPR02Medd9aMalKkUCyypDAQEXBAN3aG8WBAN3aHkZIgQg5b9IOeA+BQ71Al5EWhYUneLmb44n
CmEt5FrZ2Q3UgORUBAMxNTcFCgEFALBUoI0GAgAHCgEFANAyoI0GAgEJCAEFAADoBwQACgoC8QKe
JQoEAlvUAxsC8QIBCAMUJAAE5AAzMABOAAN6AAQwADFAAEAGrAMC8QKCIAcJowMGAAH/CAnaBgIA
P9+/z+Pr9Ph6/P4+92unzeLvt0B2vDb8Dj9j2fcGBYQisbEIkkwlkYhASDxIA8GotEwgD4bDAWCI
QB4NBYKAgf+BgQBAHcFAADgoFg0HhIKBcCGt4bDgXCYOBII//9+3y+Ps9Hf6/P5YLj4eNxePy+j1
+95/f+wMEg0JRWNByPR2NhWHQh//p5+50eRu9prwOqx6zab7ndz2/cBhcRDQfEMjE0pFctlcqEwi
DgUhoDfj09/sAer27Hc7/m9v2/wEBQOCIEJ6AkEAaCQL//39/l8Pd7Pb7vj9QQBAQBP0AAGCwYBC
G8Ig4GgoDgQAfz8fb6fX7/n+gECAcECPwAB4QAg/uEAeEAiEAaCgMAwECAHgBAMCggFg0HBAJAgp
9CQQB4OAg/+DgWBBgAHAQh8CIPBYJAwEAAAIC//oAAUDAkFAsGAQX+BQACgEJfgiEAeBCAEI/AaC
gQBAABAXoAgICgcFA4IAQnVCQQBYNAv++/j7XO4W31un0mt2233e+5Ha9oAEA9LZ3RqdU6kTyLOh
dIYsCoE9nc6XGAixAwLxAqgVCAmoAwMAX3/CgcFAyHxMLRnOR5OpuMZWJRBGwsEYYCIJAP8+3u9H
c6nI3euzOKu9rsFjuOF0e96nm+wBBwYDohEQkFAsF4wGYwFwpEQeDAXCILAQB//oAAPzAPgA+v58
/k8Pg8fl9XuAfZyePz+Ty+rzef0er1/H7fz/gIDAsHBMLh4UDIdEEklIuGMBNHGZzEWCeRB2MhUJ
A+FwgBwD+vn6/F2ujyuHwuBxOFweNweKA8Ps4nG4vJ6PY8vx/YGB4UDYfEoqF4zGg1GgCMvYWCYQ
hkLhIGAgBfr8gPz/Pl8Pj7/mA/IA+AD7gH3/3y93xAfd4gD2/f8f3/gMEg4KBsRCwZjsiAJH6SAO
hiKRIHguEAOAPl6/H2uhwdrqtPrNxx+16QIAcYeFAyIBKJxQJJBHAtEweCgK/3y9Hg63L3uy1Ok0
+w4fW9QCAKwMiYZD0jEsmkkfjQTBwKAoAfHzdvo8TebPWa3Zb/peP6gANi0ckcoFUqEogjgWCMNg
oDfTzeHn8vh7zgcrte38gYEg8JA8HAcAf36+vydzq9Lq9zzfP+gsNCUdEImlAgAEGQLxAgEIAhIk
ADOAATQAM3ABewAwsAEVADAN3gIC8QKKBQcL1QL0CUC4DwEAJAWAwCAr6AUAgAP/AgA/gC///8BA
DwASAwB4DAEgFAD/7/X+P6/o+d1zgnE/KOAAeUrAvBEB4CoEwJAUBAV0gNAYAgAoBgBgEAF/3/n+
v7fv/b+z939f8f5/3/z/4BQDAQFfAGgRAcA0BQEBHYAwBf8/z/L9r5PcOFcU7+3ABzQgEQIAMgUg
hA4BoCwFAJAOASAcCATyAUAQAAA/9ACAIAIEAQEAP7///gBAf/+AA/7/6AL/oAQBAAAEBAABAF8A
CAIAAAgAgwBAgA/wAADAgB7ACAB///v/P9/9/1///3/QIAPP/wAACAGAQAX/wAgQAfwCgAAgB9gA
/8AIAf/wAAf+/QAAB/6AAD/8EALyAMAQAwC//AGAL/wH//f/P+/8fu+v53nnCeDeaFgA0uOAmBAB
4CgEgKAOAkA4CAAgC/7/z/3+P6/k9byfbm6/PAwIAQUA/wcAAABPBQRA9oXuFAgBBACwVADQMh0W
FQbwTaLdb5YGAFBWwAABBgBQVsAACBwnJk1pY3Jvc29mdDsnV2luZG93cyA3Jzs7OzYuMS43NjAx
LjE5MTYwGiEgU1RVOydTVFUtNTMwJzsxLjAuMjswOzNHWlFTMDAwNzIbCQgwMDYyO1NUVRgHzrGb
ugU8ABULAQTVFNiQAZJCkkI=
```

The Signature Object contains the following data:

- Name
- Reason
- Date timestamp
- Application specific data
- System information
- Pen data

Signature Image

API is provided to create an image from a Signature Object. Options include:

- Image format (png/bmp etc)
- Image size
- Transparency
- EncodeData

The EncodeData option produces an image of the signature which contains the complete SignatureObject concealed in pixel attributes (using steganography), for example:

While the data is not immediately visible in the image, API allows the Signature Object to be extracted. The object can then be used for further processing, such as extracting signature capture details. Although copy and paste can be used in the normal way, care must be taken to maintain the original image contents. For example the image cannot be modified in Paint.

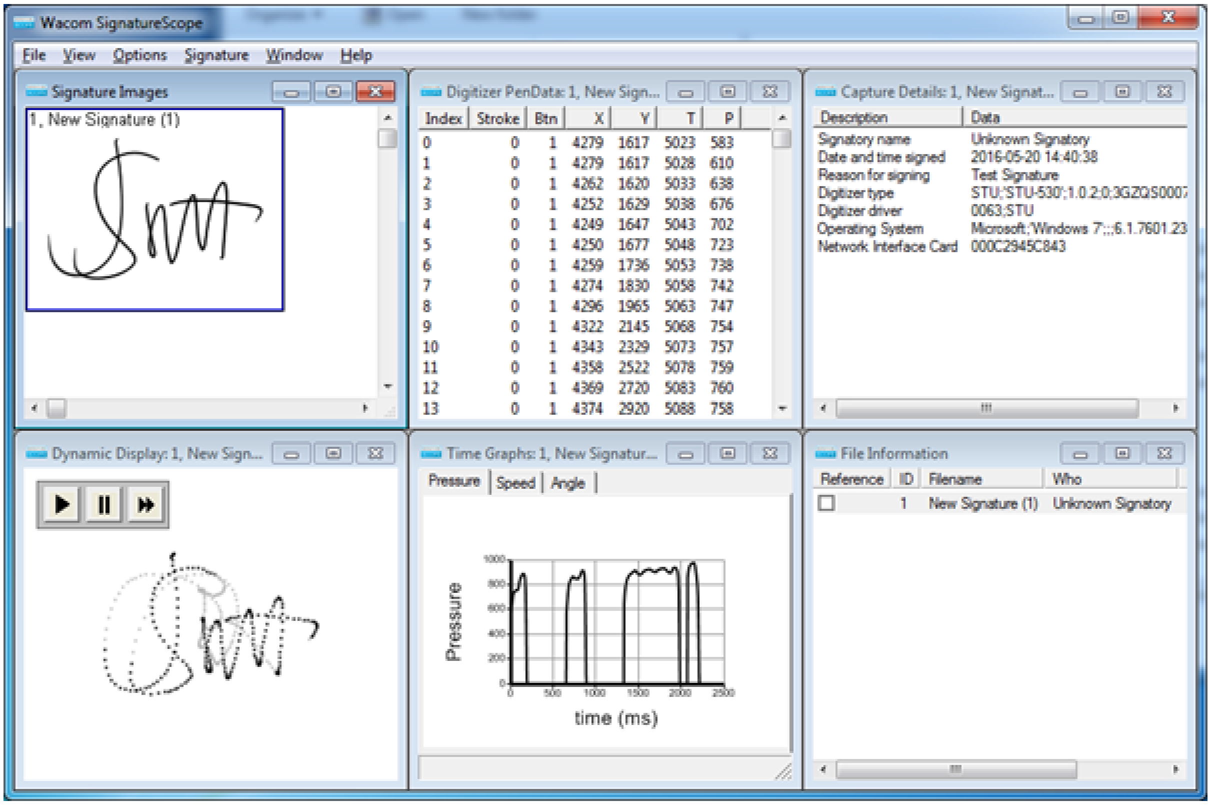
Signature Data

Data is stored in the Signature Object in an undisclosed proprietary format and API is provided to extract parts of it:

- Name
- Reason
- Date timestamp
- Application specific data

One type of application specific data is the hash of a signed document. Before starting signature capture the application can calculate a hash value for the document and include it in the signature data. At a later time the application can recalculate the hash and use the API to compare the new and saved values to determine whether any changes have been made since signing. The results can then be used to indicate the validity of a signature.

API is not provided to extract the pen data because this is personal information which could be used fraudulently. When legitimate access to the full data is needed, such as when the authenticity of a signature is assessed by a qualified Forensic Document Examiner, this is possible using the SignatureScope application:



Index	Stroke	Btn	X	Y	T	P
0	0	1	4279	1617	5023	583
1	0	1	4279	1617	5028	610
2	0	1	4262	1620	5033	638
3	0	1	4252	1629	5038	676
4	0	1	4249	1647	5043	702
5	0	1	4250	1677	5048	723
6	0	1	4259	1736	5053	738
7	0	1	4274	1830	5058	742
8	0	1	4296	1965	5063	747
9	0	1	4322	2145	5068	754
10	0	1	4343	2329	5073	757
11	0	1	4358	2522	5078	759
12	0	1	4369	2720	5083	760
13	0	1	4374	2920	5088	758

Description	Data
Signatory name	Unknown Signatory
Date and time signed	2016-05-20 14:40:38
Reason for signing	Test Signature
Digitizer type	STU;'STU-530';1.0.2;0;3GZQS0007
Digitizer driver	0063;STU
Operating System	Microsoft;'Windows 7';;;6.1.7601.23
Network Interface Card	000C2945C843

Reference	ID	Filename	Who
☐	1	New Signature (1)	Unknown Signatory

A demonstration version of SignatureScope is freely available to Signature SDK users and is fully functional with signatures captured in the application. The full version, only available to qualified document examiners, is able to import signatures for analysis.

Signature Compatibility

Signature Libraries are available for:

- Windows
- iOS

- Android

Regardless of the operating system, signatures are created in the same Signature Object format for full compatibility.
This means that signatures can be readily exchanged and used in related applications:

- SignatureScope
- Dynamic Signature Verification (DSV)
- Sign pro PDF

Signature API Examples

The Signature Library includes examples in a range of languages including:

- HTML
- Java
- JavaScript
- C++
- C#
- VB

A full description of the API can be found in the Signature-Library-API document.

The following examples demonstrate use of the API in Javascript:

Capture a signature:

Capture

```
sigCtl  = new ActiveXObject("Florentis.SigCtl");

dynCapt = new ActiveXObject("Florentis.DynamicCapture");

rc = dynCapt.Capture(sigCtl,"Who","Why");
```

Following successful capture, save the text format of the Signature Object:

Save SigText

```
SignatureText = sigCtl.Signature.SigText; // save Base64 encoded signature
```

Create a PNG image of the signature:

Create PNG

```
rc = sigCtl.Signature.RenderBitmap(filename, 300, 150, "image/png", 0.5,
                                      0xff0000, 0xffffff, 0.0, 0.0, flags );
```

In a Windows Forms application it is not necessary to create an image file to display the signature. The ActiveX component provides automatic rendering of the signature and this is demonstrated in the HTML samples for Internet Explorer:

Auto render

```
...
   var sigCtl = document.getElementById("sigCtl1");
   sigCtl.Signature.SigText=SignatureText;
...


 <div>
   <object id="sigCtl1" style="width:60mm;height:35mm"
           type="application/x-florentis-signature">
   </object>
 </div>
```

ActiveX support has been removed from new versions of Internet Explorer and is generally not available in alternative browsers. For this reason SigCaptX is available as an extension of the Signature Library to give web browser applications full access to the Signature Library API.

Pen Tablet Devices

The Signature Library detects the type of Wacom hardware connected and uses the appropriate pen interface.

There are two distinct types of pen device supported:

- Signature tablet – range of STU tablets
- Pen tablets and displays – range of Intuos, DTU and Cintiq tablets

The STU tablets have a dedicated USB interface for pen input.
The pen displays and pen tablets are installed with a driver. The driver provides a Wintab interface which is used for signature capture.

Signature Library Installation

The Signature Library is installed as a set of ActiveX/COM components in the folder:

```
C:\Program Files\WacomGSS
```

When the 32-bit version is installed on 64-bit Windows the components are installed in:

```
C:\Program Files (x86)\WacomGSS
```

The installation requires COM registration with a number of supporting entries in the Windows registry and is achieved by running the installer supplied in the SDK download package.