DigitalPersona, Inc.

Upgrading Platinum SDK Applications to One Touch® for Windows® SDK

C# Edition

Version 1.6.1

Developer Guide

Should you have any questions concerning this document, or if you need to contact DigitalPersona for any other reason, write to:

DigitalPersona, Inc.

720 Bay Road

Suite 100

Redwood City, CA 94063
USA

Document Publication Date: December 3, 2010 (1.6.1)

Table of Contents

Introduction 1

The purpose of this document is to support programmers during their conversion from using the Platinum SDK to the new One Touch for Windows ("OTW") SDK. It is intended to bridge the gap between the two API's and to act as an accompaniment to the OTW SDK guide and sample code that are included with the OTW SDK. Only the most used API calls are included in this document.

This document includes an example of the new approach in which to subscribe to fingerprint reader events. The document also contains a comparison of Platinum code snippets for each component area (i.e., capture, enrollment, verification) with a similar OTW code snippet. Also included is a code snippet of how to convert a Platinum fingerprint template to an OTW fingerprint template. A mapping between new and old fingerprint index enumerations is given to emphasize the importance of converting these indexes.

For questions, comments, and concerns, please email TechSupport@DigitalPersona.com. We at DigitalPersona extend our appreciation for your business.

Event Subscription

Use Code Sample 1 in applications developed in Microsoft C# to handle DPFP.Capture.Capture events. Event handling for DPFP.Gui.Enrollment and DPFP.Gui.Verification is also achieved using inheritance of those classes.

Code Sample 1. Handle capture events in C#:

```
/* NOTE: This form is inherited from the CaptureForm,
   so the VisualStudio Form Designer may not load it properly
   (at least until you build the project).
   If you want to make changes in the form layout - do it in the base
  CaptureForm.
   All changes in the CaptureForm will be reflected in all derived forms
   (i.e. in the EnrollmentForm and in the VerificationForm)
 */
    public class CaptureForm : DPFP.Capture.EventHandler {

            // Prepare to capture.
            public void CaptureForm() {

                // Create a capture operation.
                Capturer = new DPFP.Capture.Capture();

                // Subscribe for capturing events.
                Capturer.EventHandler = this;

                // Start capture.
                Capturer.StartCapture();
            }

            public void OnComplete(object Capture, string SerNum, DPFP.Sample Sample) { … }
            public void OnFingerGone(object Capture, string SerNum) { … }
            public void OnFingerTouch(object Capture, string SerNum) { … }
            public void OnReaderConnect(object Capture, string SerNum) { … }
            public void OnReaderDisconnect(object Capture, string SerNum) { … }
            public void OnSampleQuality(object Capture, string SerNum,
            DPFP.Capture.CaptureFeedback CaptureFeedback) { … }
    }
```

Sample Conversion

Use DPFP.Capture.SampleConversion.ConvertToPicture(Sample, ref Bitmap) to return a Bitmap.

Supported Platinum SDK Libraries and Components

FPDevices

OTW SDK Equivalent
DPFP.Capture.ReadersCollection

Public Members

Platinum SDK
Count
OTW SDK Equivalent
Count
Item
OTW SDK Equivalent
Item
DeviceConnected(), DevDisconnected()
OTW SDK Equivalent
See DPFP.Capture.EventHandler

FPDevice

OTW SDK Equivalent
DPFP.Capture.ReaderDescription

Public Members

Platinum SDK
Language
OTW SDK Equivalent
Language

Vendor

OTW SDK Equivalent

Vendor

Product

OTW SDK Equivalent

Product

SerialNumber

OTW SDK Equivalent

SerialNumber

HWRevision

OTW SDK Equivalent

HardwareRevision

FWRevision

OTW SDK Equivalent

FirmwareRevision

Type

OTW SDK Equivalent

SerialNumberType

ImageType

OTW SDK Equivalent

ImpressionType

FingerTouching(), FingerLeaving(), SampleAcquired()

OTW SDK Equivalent

See DPFP.Capture.EventHandler

Error([in] AIErrors errcode)

OTW SDK Equivalent

See DPFP.Error.SDKException

FPSample

OTW SDK Equivalent

DPFP.Sample

Public Members

Platinum SDK

Export([out] VARIANT* pVal, [out, retval] AIErrors *pErr)

OTW SDK Equivalent

Serialize()

Import([in] VARIANT Val, [out, retval] AIErrors *pErr)

OTW SDK Equivalent

Deserialize(byte[] Template)

Width, Height, PictureWidth, PictureHeight, Picture

OTW SDK Equivalent

Use DPFP.Capture.SampleConversion.ConvertToPicture(Sample, ref Bitmap) to create a Bitmap.

FPTemplate

OTW SDK Equivalent

DPFP.Template, DPFP.FeatureSet

Use DPFP.Template if .TemplType is Tt_PreRegistration.

Use DPFP.FeatureSet if.TemplType is Tt_Verification.

Public Members

Platinum SDK

Export([out] VARIANT* pVal, [out, retval] AIErrors *pErr)

OTW SDK Equivalent

Serialize()

Import([in] VARIANT Val, [out, retval] AIErrors *pErr)

OTW SDK Equivalent

Deserialize(byte[] Template)

TemplData

OTW SDK Equivalent

Bytes

FPFtrEx

OTW SDK Equivalent

DPFP.Processing.FeatureExtraction

Public Members

Platinum SDK

Process([in]IDispatch *pSample, [in] AITemplateTypes TemplType,
[out] IDispatch **ppTemplate, [out] AISampleQuality **pQuality,
[out, retval] AIErrors** pErr)

OTW SDK Equivalent

CreateFeatureSet(Sample, DataPurpose, ref CaptureFeedback, ref FeatureSet)

FPRegister

OTW SDK Equivalent

DPFP.Processing.Enrollment

Public Members

Platinum SDK

NewRegistration([in] AIRegTargets Target, [out,retval] AIErrors *pErr)

OTW SDK Equivalent

Instantiate a new DPFP.Processing.Enrollment object, or use Clear() if the object is already instantiated.

Add([in] IDispatch *pPreReg, [out] VARIANT_BOOL *pbDone, [out,retval] AIErrors *pErr)

OTW SDK Equivalent

AddFeatures(FeatureSet)

Note: Use the FeaturesNeeded and TemplateStatus properties instead of the pDone and pErr out parameters.

RegistrationTemplate

OTW SDK Equivalent

Template

FPVerify

OTW SDK Equivalent

DPFP.Verification

Public Members

Platinum SDK

Compare([in] IDispatch *pTemplate, [in] Idispatch *pVerTemplate,
[out] VARIANT_BOOL *pVerifyOk, [out] VARIANT *pScore,
[out] VARIANT *pThreshold, [out] VARIANT_BOOL *pLearnDone,
[out] IsecureModeMask *pSecurity, [out, retval] AIErrors *pErr)

OTW SDK Equivalent

Verify(FeatureSet, Template, ref Result)

SecurityLevel

OTW SDK Equivalent

FARRequested

FPGetSample, FPGetSampleX

OTW SDK Equivalent

DPFP.Capture.Capture

Public Members

Platinum SDK

CreateOp([out,retval] AIErrors *pErr), Run([out,retval] AIErrors *pErr)

OTW SDK Equivalent

StartCapture()

Cancel([out,retval] AIErrors *pErr)

OTW SDK Equivalent

StopCapture()

SetDevicePriority([in] AIDevPriorities Priority, [in] LONG hWnd, [out,retval] AIErrors *pErr), SelectDevice([in] BSTR serNum, [out,retval] AIErrors
*pErr)

OTW SDK Equivalent

Priority

Done([in] IDispatch *pTemplate), DevDisconnected(), DeviceConnected()

OTW SDK Equivalent

See DPFP.Capture.EventHandler

Error([in] AIErrors errcode)

OTW SDK Equivalent

See DPFP.Error.SDKException

FPGetTemplate

See FPRegisterTemplate, FPVerifyTemplate

FPGetTemplateX

See FPRegisterTemplateX, FPVerifyTemplateX

FPRegisterTemplate, FPRegisterTemplateX

OTW SDK Equivalent

DPFP.Enrollment, DPFP.Gui.Enrollment

Public Members

Platinum SDK

CreateOp([out,retval] AIErrors *pErr), Run([out,retval] AIErrors *pErr), Cancel([out,retval] AIErrors *pErr), SetDevicePriority([in] AIDevPriorities Priority, [in] LONG hWnd, [out,retval] AIErrors *pErr), SelectDevice([in] BSTR serNum, [out,retval] AIErrors *pErr)

OTW SDK Equivalent

See DPFP.Capture.Capture

Done([in] IDispatch *pTemplate), DevDisconnected(), DeviceConnected(), SampleReady(), SampleQuality()

OTW SDK Equivalent

See DPFP.Capture.EventHandler

Error([in] AIErrors errcode)

OTW SDK Equivalent

See DPFP.Error.SDKException

FPVerifyTemplate, FPVerifyTemplateX

OTW SDK Equivalent

DPFP.Verification, DPFP.Gui.Verification

Public Members

Platinum SDK

CreateOp([out,retval] AIErrors *pErr), Run([out,retval] AIErrors *pErr), Cancel([out,retval] AIErrors *pErr), SetDevicePriority([in] AIDevPriorities Priority, [in] LONG
hWnd, [out,retval] AIErrors *pErr), SelectDevice([in] BSTR serNum, [out,retval] AIErrors
*pErr)

OTW SDK Equivalent

See DPFP.Capture.Capture

SecurityLevel

OTW SDK Equivalent

FARRequested

Done([in] IDispatch *pTemplate), DevDisconnected(), DeviceConnected(), SampleReady(), SampleQuality()

OTW SDK Equivalent

See DPFP.Capture.EventHandler, DPFP.Gui.Verification.EventHandler

Error([in] AIErrors errcode)

OTW SDK Equivalent

See DPFP.Error.SDKException

SDK Sample Code Comparison

This section illustrates how capture, enrollment and verifications functions are performed in both Platinum and OTW. Sample code using the Platinum SDK is displayed to the left, and sample code using OTW SDK is displayed to the right.

Capture

Platinum SDK	OTW SDK Equivalent

`private DPSDKOPSLib.FPRegisterTemplateClass rgs = new DPSDKOPSLib.FPRegisterTemplateClass();`
 	
`private void cmd_begin_Click(object sender, System.EventArgs e)`
`{`
`rgs.Done += new DPSDKOPSLib._IFPRegisterTemplateEvents_DoneEventHandler(Done);`
`rgs.Run(1);`
`}`
 	
`private void cmd_exit_Click(object sender, System.EventArgs e)`
`{`
`rgs.Cancel();`
`Application.Exit();`
`}`
 	
`private void Done(object obj)`
`{`
`DpSdkEngLib.FPTemplate tmplate = (DpSdkEngLib.FPTemplate)obj;`
`tmplate.Export(ref pRegTmplate);`
`}`
	`/* NOTE: This form is a base for the EnrollmentForm and the VerificationForm. All changes in the CaptureForm will be reflected in all its derived forms.`
	`*/`
	`public partial class CaptureForm : Form, DPFP.Capture.EventHandler`
	`{`
	`private void CaptureForm_Load(object sender, EventArgs e)`
	`{`
	`Capturer = new DPFP.Capture.Capture(); // Create a capture operation.`
	`Capturer.EventHandler = this; // Subscribe for capturing events.`
	`Capturer.StartCapture();`
	`}`
	
	`private void CaptureForm_FormClosed(object sender, FormClosedEventArgs e) { Capturer.StopCapture(); }`
	
	`public void OnComplete(object Capture, string ReaderSerialNumber, DPFP.Sample Sample) { Process(Sample);. }`
	`}`

Enrollment

Platinum SDK

```
private DPSDKOPSLib.FPRegisterTemplateClass rgs = new
DPSDKOPSLib.FPRegisterTemplateClass();

private void cmd_begin_Click(object sender, System.EventArgs e)
  {
    rgs.Done += new
DPSDKOPSLib._IFPRegisterTemplateEvents_DoneEventHandler(RegDone);
    cmd_begin.Enabled = false;
     rgs.Run(1);
  }

private void Done(object obj)
  {
   DpSdkEngLib.FPTemplate  tmplate = (DpSdkEngLib.FPTemplate)obj;
   tmplate.Export(ref pRegTmplate);
  }
```

OTW SDK Equivalent

```
public class EnrollmentForm : CaptureForm
{
 private DPFP.Processing.Enrollment Enroller;
 public delegate void OnTemplateEventHandler(DPFP.Template template);
  public event OnTemplateEventHandler OnTemplate;

  protected override void Init()
  {
    Enroller = new DPFP.Processing.Enrollment();    // Create an enrollment.
  }

  protected DPFP.FeatureSet ExtractFeatures(DPFP.Sample Sample,
DPFP.Processing.DataPurpose Purpose)
  {
   DPFP.Processing.FeatureExtraction Extractor = new
DPFP.Processing.FeatureExtraction(); // Create a feature extractor
    DPFP.Capture.CaptureFeedback feedback = DPFP.Capture.CaptureFeedback.None;
    DPFP.FeatureSet features = new DPFP.FeatureSet();
    Extractor.CreateFeatureSet(Sample, Purpose, ref feedback, ref features);
    if (feedback == DPFP.Capture.CaptureFeedback.Good)
     return features;
    else
      return null;
 }

protected override void Process(DPFP.Sample Sample)
  {
  // Process the sample and create a feature set for the enrollment purpose.
  DPFP.FeatureSet features = ExtractFeatures(Sample,
DPFP.Processing.DataPurpose.Enrollment);

  // Check quality of the sample and add to enroller if it's good
  if (features != null) try
  {
   MakeReport("The fingerprint feature set was created.");
    Enroller.AddFeatures(features);   // Add feature set to template.
  }
  finally {

   // Check if template has been created.
   switch(Enroller.TemplateStatus)
   {
    case DPFP.Processing.Enrollment.Status.Ready: // report success and stop capturing
     Capturer.StopCapture();
     break;

     case DPFP.Processing.Enrollment.Status.Failed: // report failure and restart
```

	capturing Enroller.Clear(); Capturer.StopCapture(); Capturer.StartCapture(); break; } } }

Verification

Platinum SDK

```
private DPSDKOPSLib.FPGetTemplateClass verifyTemple = new
DPSDKOPSLib.FPGetTemplateClass();

private void cmd_begin_Click(object sender, System.EventArgs e)
   {
     verifyTemple.Done +=  new
DPSDKOPSLib._IFPGetTemplateEvents_DoneEventHandler(VerifyDone);
     verifyTemple.Run(0);
   }

private void Done(object vt)
   {
     bool verifyOK = false;
     object score = 0;
     object threshold = 0;
     bool tture = true;
     DpSdkEngLib.FPTemplate  verifyTemplate =  (DpSdkEngLib.FPTemplate)vt;
     DpSdkEngLib.FPTemplateClass  RegTemplate =  new
DpSdkEngLib.FPTemplateClass();

     RegTemplate.Import(cpRegTmplate);
     DpSdkEngLib.FPVerifyClass verify = new DpSdkEngLib.FPVerifyClass();
    verify.Compare(RegTemplate,verifyTemplate, ref verifyOK, ref score, ref threshold,
ref tture, ref sm);
     if (verifyOK == true)
      this.helptext.Text = "Finger print matched!!";
     else
      this.helptext.Text = "Finger print did not match !!";
   }
  }
}
```

OTW SDK Equivalent

```
 public class VerificationForm : CaptureForm
 {
   private DPFP.Template Template;
   private DPFP.Verification.Verification Verificator;

   protected override void Init()
   {
     Verificator = new DPFP.Verification.Verification();  // Create a fingerprint template
verificator
   }

   protected override void Process(DPFP.Sample Sample)
   {
     // Process the sample and create a feature set for the enrollment purpose.
     DPFP.FeatureSet features = ExtractFeatures(Sample,
DPFP.Processing.DataPurpose.Verification);

     // Check quality of the sample and start verification if it's good
     if (features != null)
     {
       // Compare the feature set with our template
       DPFP.Verification.Verification.Result result = new
DPFP.Verification.Verification.Result();
       Verificator.Verify(features, Template, ref result);
       if (result.Verified) MakeReport("The fingerprint was VERIFIED.");
       else  MakeReport("The fingerprint was NOT VERIFIED.");
     }
  }
}
```

Platinum SDK Enrollment Template Conversion for C#

Use Code Sample 2 in applications developed in Microsoft C# to convert Platinum SDK enrollment templates.

Code Sample 2. Platinum SDK Template Conversion for C#:

```
private byte[] PlatinumToOTW(object platinumTemplate)
  {
    FPTemplate pTemplate = new FPTemplate;
    DpSdkEngLib.AIErrors er = pTemplate.Import(platinumTemplate);
    if (er != DpSdkEngLib.AIErrors.Er_OK)
    {
      return null;
    }
    return (byte[])pTemplate.TemplData;
  }
```

Data Type Conversion

AIFinger

Platinum SDK	OTW SDK Equivalent
Fn_LeftPinkie = 0	10 = Left little finger
Fn_LeftRing = 1	9 = Left ring finger
Fn_LeftMiddle = 2	8 = Left middle finger
Fn_LeftIndex = 3	7 = Left index finger
Fn_LeftThumb = 4	6 = Left thumb
Fn_RightThumb = 5	1 = Right thumb
Fn_RightIndex = 6	2 = Right index finger
Fn_RightMiddle = 7	3 = Right middle finger
Fn_RightRing = 8	4 = Right ring finger
Fn_RightPinkie = 9	5 = Right little finger

Index