Investintech.com Inc.
Software Development Kit:
ImagetoPDF Function Library
User's Guide

December 31, 2007
http://www.investintech.com

Copyright 2007 Investintech.com, Inc. All rights reserved

Adobe® is registered by Adobe Systems Incorporated
Acrobat® is registered by Adobe Systems Incorporated
Postscript® is registered by Adobe Systems Incorporated
Access™ is registered by Microsoft Corporation
Excel® is registered by Microsoft Corporation
Visual Basic® is registered by Microsoft Corporation
Visual C++® is registered by Microsoft Corporation
Visual C#® is registered by Microsoft Corporation
Visual J#® is registered by Microsoft Corporation
Visual Studio® is registered by Microsoft Corporation
Win32® is registered by Microsoft Corporation
Windows® is registered by Microsoft Corporation
Windows NT® is registered by Microsoft Corporation
Windows Vista™ is registered by Microsoft Corporation

Contents

Preface **1**
About This Documentation 1
Typographical Conventions Used in This Document 1
Getting More Information 2
Customer Service and Technical Support 2
Fax and Mailing Address 2
Investintech ImagetoPDF Conversion DLL **3**
What is the Investintech ImagetoPDF Conversion DLL? 4
Installation Instructions for ImagetoPDFDLL 4
System Requirements 4
Starting the Installation 4
Using the Investintech ImagetoPDF Conversion DLL in Microsoft Visual C++ .NET 2003 11
Implicit Linking 11
Linking .lib file with project 12
Using methods from DLL file 13
Using the Investintech ImagetoPDFDLL Conversion DLL in Microsoft Visual C++ .NET 2003 13
Investintech ImagetoPDFDLL Conversion DLL Methods 14
Interface 14
Parameter Type 14
File Names 15
Error Handling 15
Common Sample Source Code 15
Conversion from Image file to PDF document 16
Image_to_PDF _CBR 16
Prototype 16
Description 16
Calling Convention 16
Parameters 16
Returns 16
Example 16
Image_to_PDF_CBT 17
Prototype 17
Description 17
Calling Convention 17
Parameters 17
Returns 17
Example 17
Image_to_PDF_CLR 17
Prototype 17
Description 18
Calling Convention 18
Parameters 18
Returns 18
Example 18
Image_to_PDF_CLT 18

Prototype 18
Description 18
Calling Convention 18
Parameters 18
Returns 19
Example 19
Image_to_PDF_SBR 19
Prototype 19
Description 19
Calling Convention 19
Parameters 19
Returns 20
Example 20
Imahe_to_PDF_SBT 20
Prototype 20
Description 20
Calling Convention 20
Parameters 20
Returns 20
Example 20
Image_to_PDF_SLR 21
Prototype 21
Description 21
Calling Convention 21
Parameters 21
Returns 21
Example 21
Image_to_PDF_SLT 21
Prototype 21
Description 22
Calling Convention 22
Parameters 22
Returns 22
Example 22
VB6 callable code 23
Image_to_PDF_VB6 23
Prototype 23
Description 23
Calling Convention 23
Parameters 23
Returns 23
Example 23
InvestintechConversionDLL_IDString 24
Description 24
Parameters 24
Returns 24
Investintech ImagetoPDF Conversion COM Server Methods 25
ImagetoPDF 25
Prototype 25
Description 25
Parameters 25
Returns 25

Example 25
Index 26

Preface

Welcome to Investintech.com Inc. Software Development Kit (SDK). This SDK provides technologies for converting image files into PDF files. This user's guide provides the basic information needed to use all parts of the SDK. This SDK consists of:

- Investintech Image to PDF Conversion DLL, a Dynamic-Link Library for converting PDF files

This document provides a comprehensive description of the DLL.

About This Documentation

This documentation consists of:

- Investintech Image to PDF Conversion DLL, describing usage and methods of DLL designed to create PDF files from various image formats (BMP, GIF, JPG, PNG, TIF).

Typographical Conventions Used in This Document

The following table describes typographical and naming conventions used in this document:

Font	*Used for*	*Examples*
`monospaced`	Expected input from the user	`imagetopdf -iabc.bmp -oabc.pdf`
	Paths and filenames	`c:\investintech\imagetopdf.exe`
	Source code	`MessageBox("ABC");`
`monospaced blue`	Source code keywords	`If`
`monospaced green`	Source code comments	`//this is a comment`
`monospaced bold`	Method name	The **`AboutBox()`** method
'**Single quoted bold**'	Elements of the user interface like buttons, icons etc.	click on '**Start**' button, press '**Enter**'
blue underlined	Internet links	http://www.investintech.com

Getting More Information

Investintech.com Inc. conducts research in data conversion technology and develops data conversion products utilizing results of the research. More information about our company is available at the company's website http://www.investintech.com.

Customer Service and Technical Support

Investintech.com Inc. strives to provide the best possible technical support to its customers and prospective customers. If you would like personal assistance, please feel free to call us or send e-mail to Customer Service or Technical Support. We are available by phone during regular business hours and in the vast majority of cases, we will return our e-mail with an answer the same day it is received.

	Customer Service	*Technical Support*
Telephone:	+1 416 920 5884	+1 416 920 2539
E-mail:	cs@investintech.com	techsupport@investintech.com
Hours:	Our Business Hours are Monday to Friday 9am-6pm (Eastern Time GMT-5:00).	

Fax and Mailing Address

Mailing Address:	Investintech.com Inc. 600 - 425 University Avenue Toronto, ON M5G 1T6 Canada
Fax:	+1 416 920 5848

Investintech.com Inc. Software Development Kit User's Guide

Investintech ImagetoPDF Conversion DLL

What is the Investintech ImagetoPDF Conversion DLL?

The Investintech ImagetoPDF Conversion DLL is a collection of methods compiled, linked and stored in a dynamic-link library (DLL) file. These methods provide a set of methods for converting files from the supported image formats (BMP, GIF, JPG, PNG, TIF) to PDF format.

Installation Instructions for ImagetoPDFDLL

Below you will find step-by-step instructions to install ImagetoPDFDLL Conversion Library on your system. You may want to print this instruction sheet for reference before beginning to install ImagetoPDFDLL.

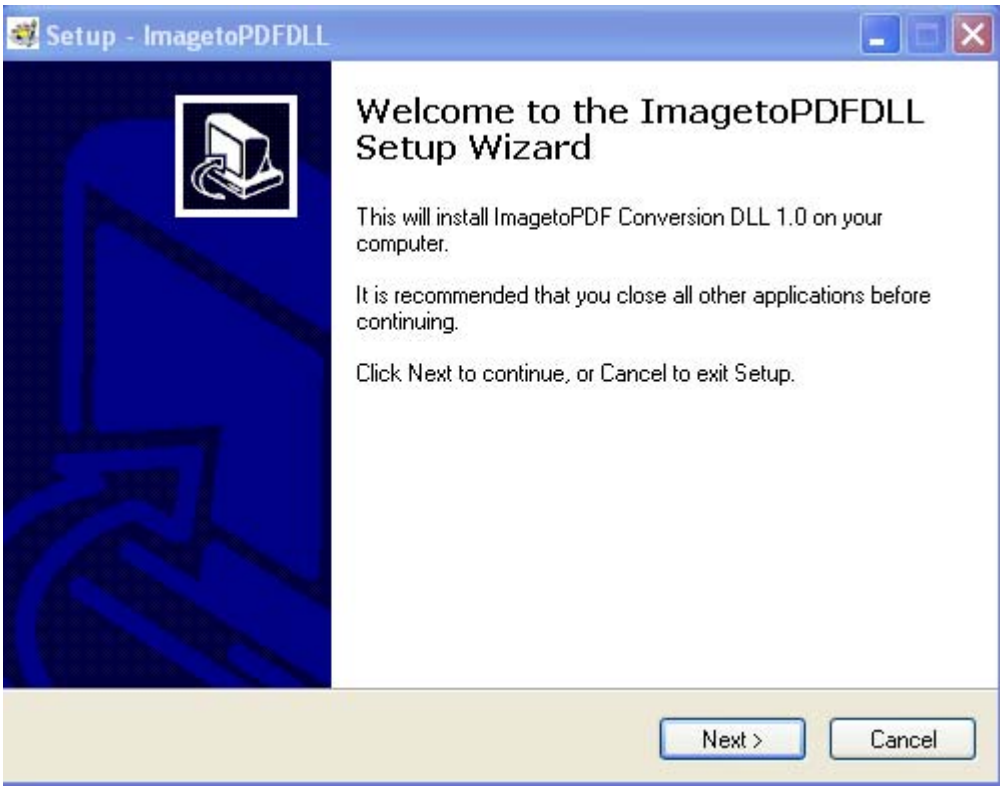
System Requirements

The minimum computer system resources required to install and use PDF2ExcelDLL are:

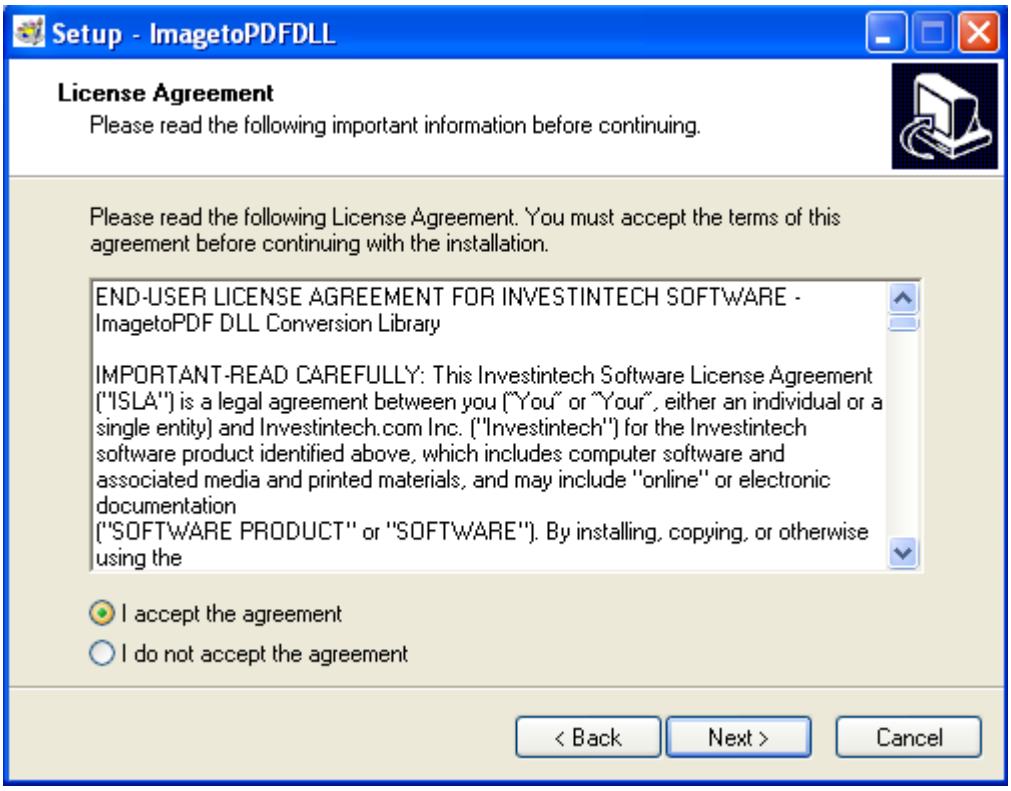
- Microsoft Windows XP
- 64 MB of RAM
- 10 MB of available hard disk space
- A software development environment, such as Microsoft Visual Studio or Borland Delphi

Starting the Installation

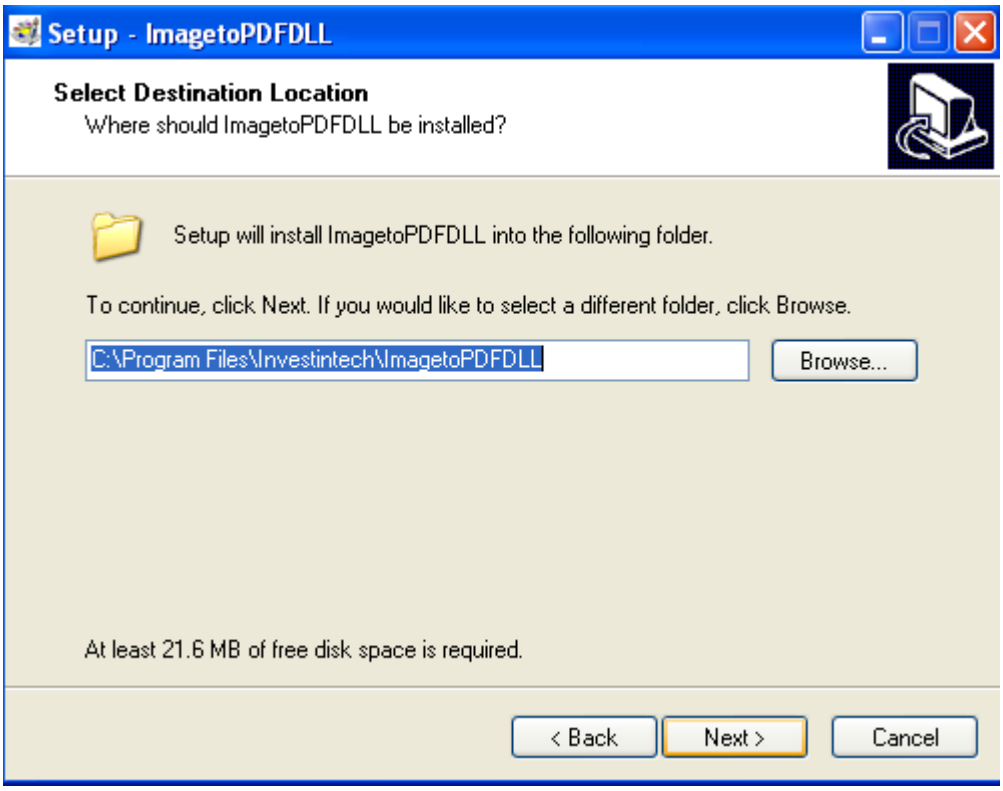
Start the installation by launching the `InstallImagetoPDFDLL.exe` file, e.g. by double clicking it in the Windows Explorer. After that you should see a window that looks like this:



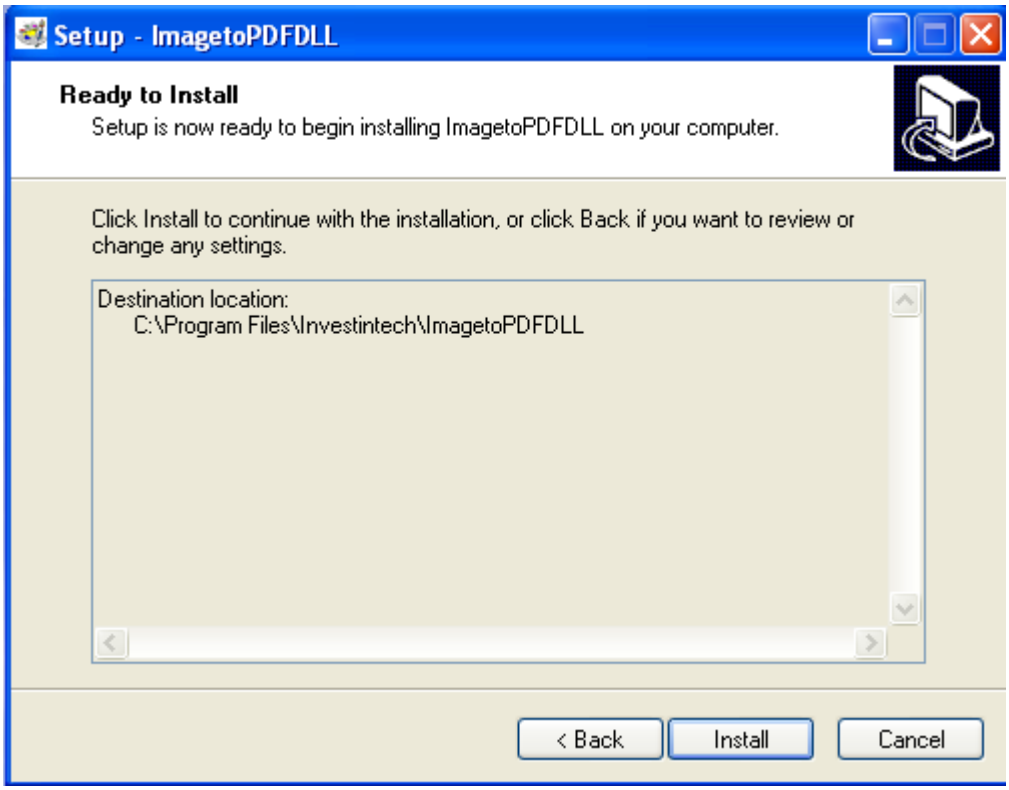
Click the '**Next**' button to continue installation. You should see the next window:



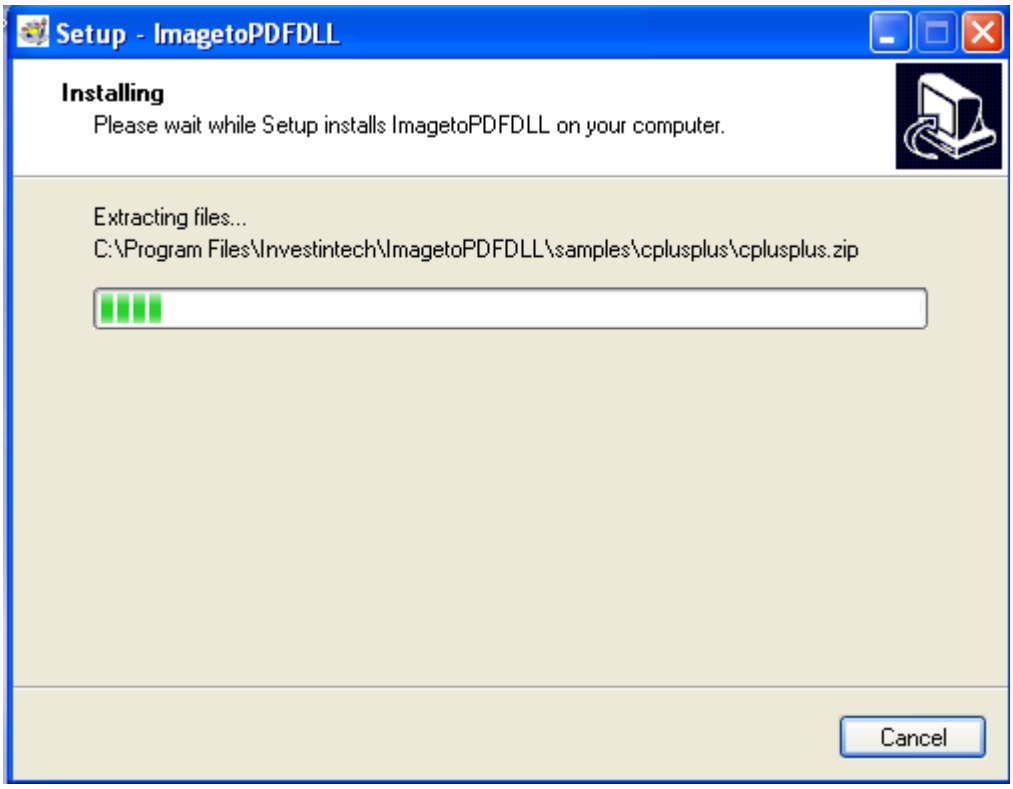
Please read the Investintech Software License Agreement carefully and thoroughly. If you wish to continue with installation, choose '**I agree with the above terms and conditions**' and click '**Next**'. Otherwise, click '**Exit**' and installation will terminate. If you clicked '**Next**' you should see the next window:



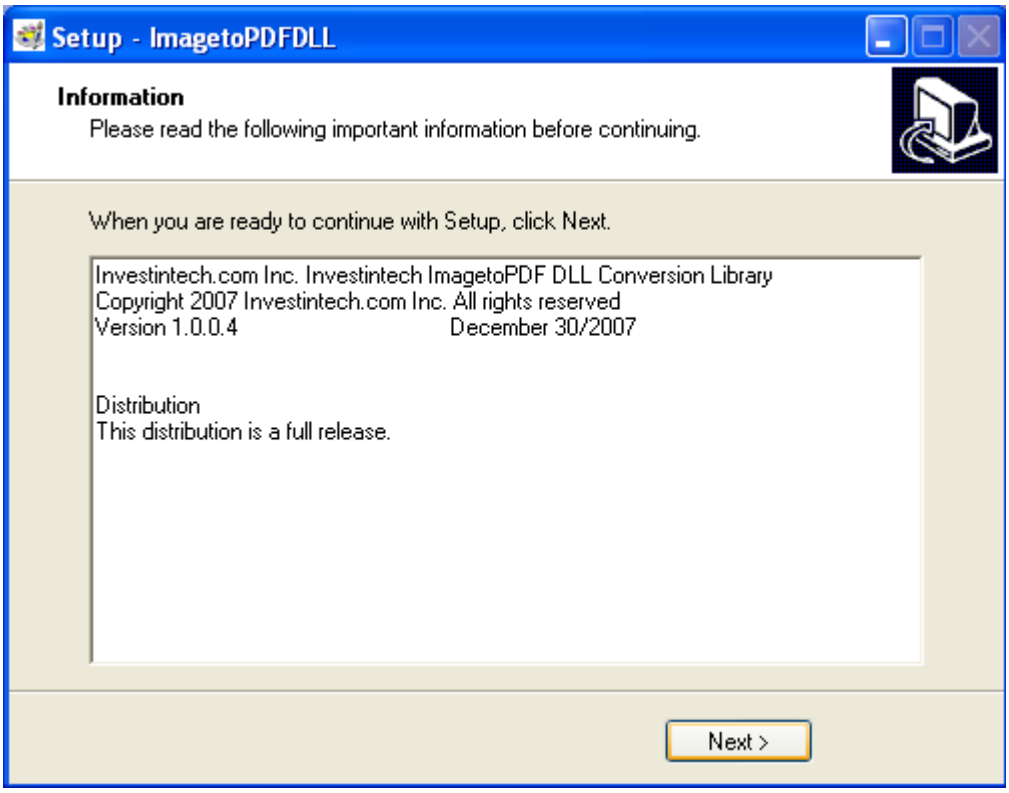
On this screen you can choose the installation directory for the ImagetoPDFDLL software. The default installation folder will be contained in the `Program Files` folder on your system disk. It is possible to change the location of installation folder either by typing its path directly or by clicking the 'Browse**...**' button. If the destination folder (directory) you have chosen does not exist, the installation program will create it. Click the '**Next**' button to continue the installation.



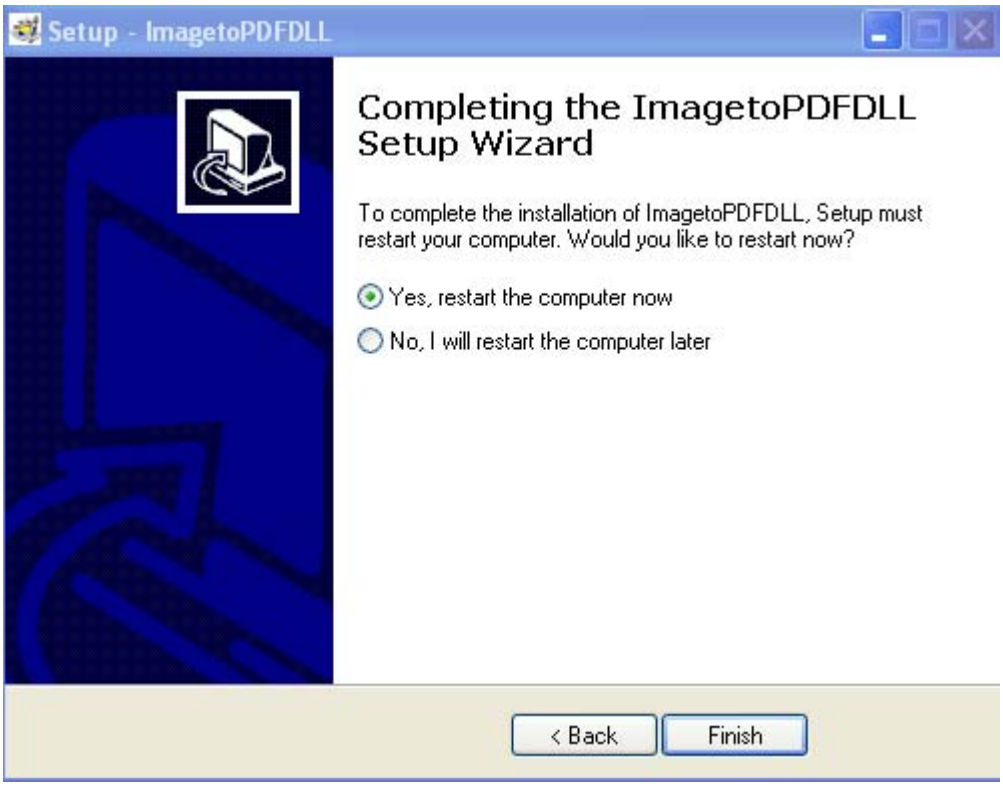
Clicking the '**Install**' button will start the installation.



The installation will copy the files that comprise ImagetoPDFDLL to the directory you specified earlier. After a very short period of time, usually about 30 seconds, you should see the next window:



This windows describes the distribution of ImagetoPDFDLL being installed. Click '**Next**' to continue.



When this window appears, the installation process is complete. Click '**Finish**' to end the installation program.

Using the Investintech ImagetoPDF Conversion DLL in Microsoft Visual C++ .NET 2003

In this section you will learn how to use the Investintech ImagetoPDF Conversion DLL. All pictures and examples from now on will be based on Microsoft Visual C++ .NET 2003.

In order to use methods contained in DLL you have to link an executable file to DLL. Executable file can be either EXE or DLL file. There are two possible ways of linking: implicit and explicit linking. In this documentation implicit linking will be shown.

Implicit Linking

Implicit linking is load-time linking. The following is needed for executables to implicitly link to Investintech Conversion DLL:

- ImagetoPDF`DLL.h`, the header file which contains declarations of exported methods
- ImagetoPDF`DLL.lib`, an import library used by the linker

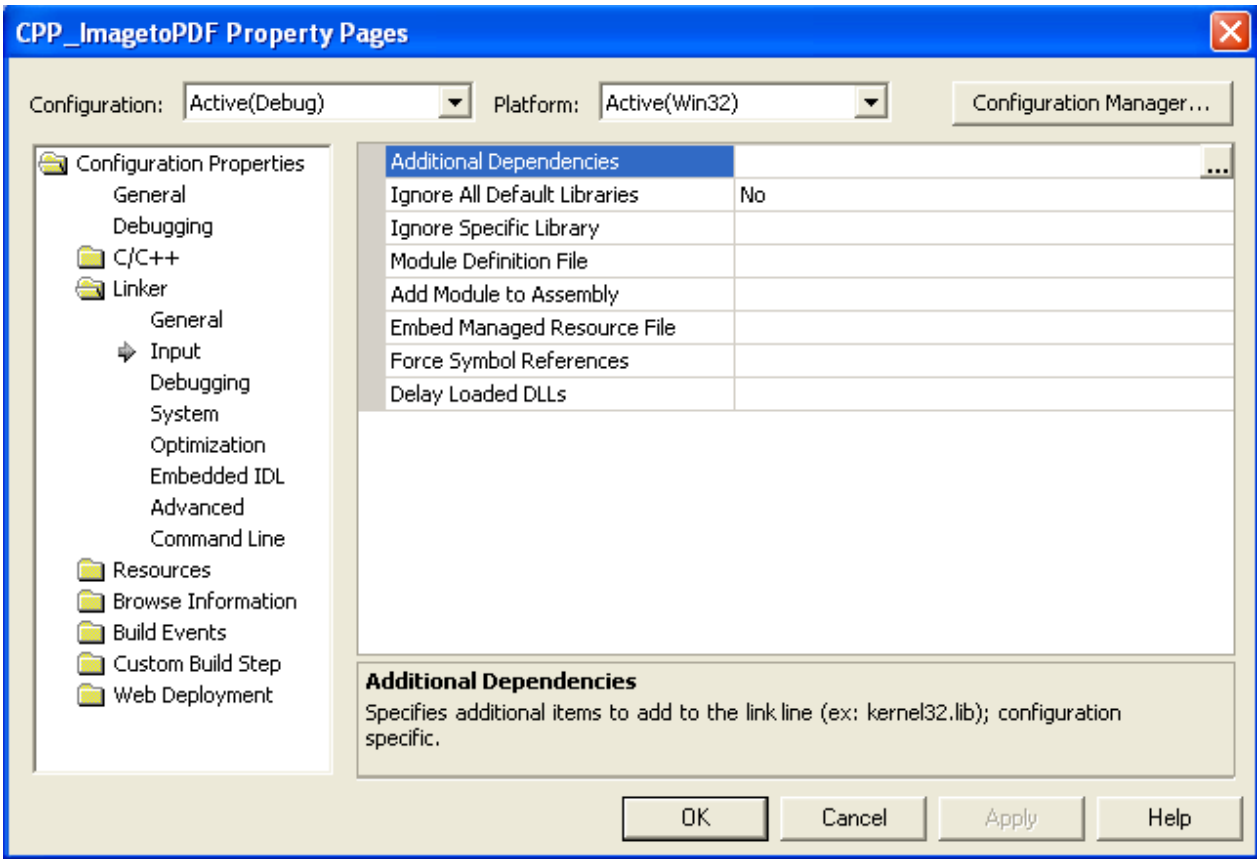
- ImagetoPDFDLL`.dll`, the Dynamic Link Library file

All these files are provided by Investintech.com Inc.

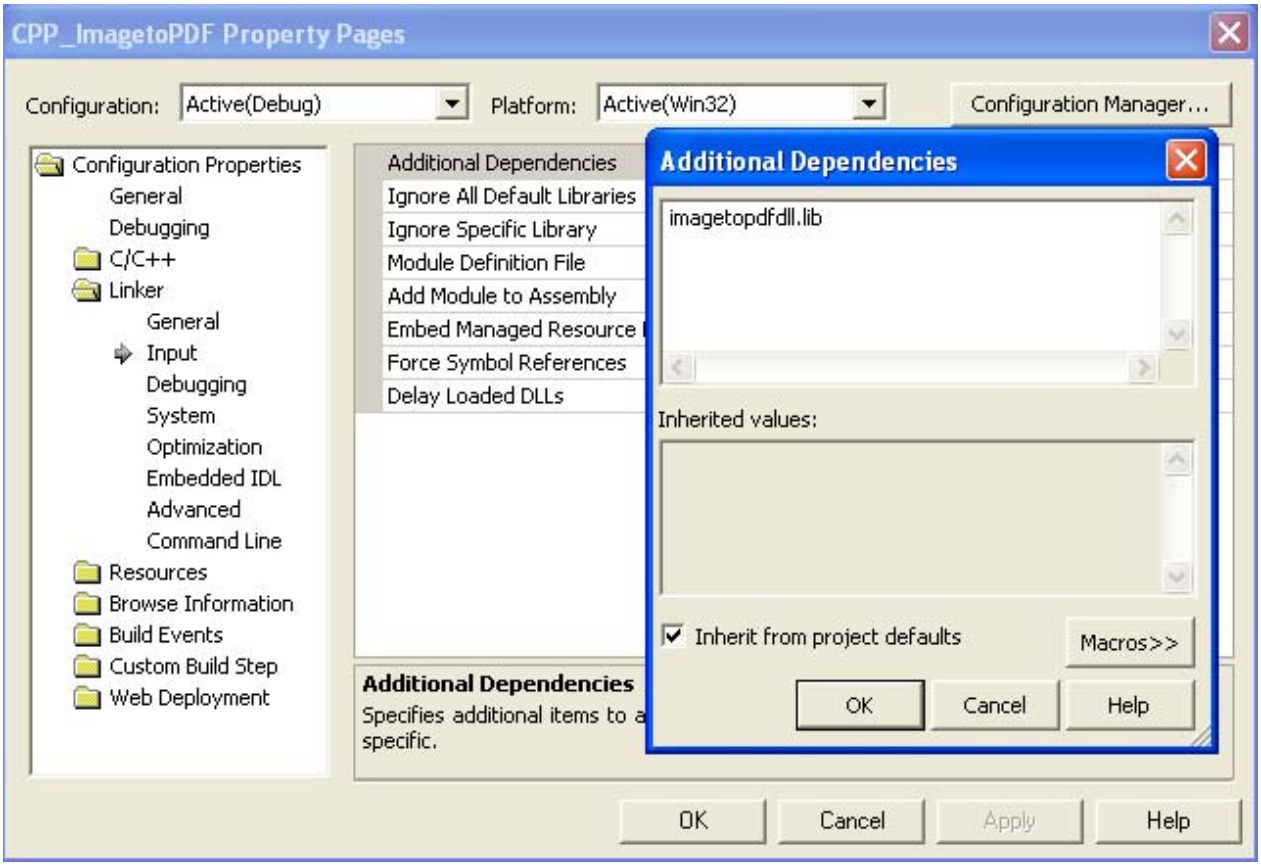
Linking .lib file with project

Below you will find step-by-step instructions on how to link ImagetoPDFDLL`.lib` file with your project.

Start Visual Studio and select '**File** > **Open** > **Project**'. Select the project in which you want to use ImagetoPDFDLL`.dll` and click '**OK**'. Select '**Project** > **Properties**'. You should see the window like next one:



Click the '**Linker**' folder and open '**Input**' property page. Modify the '**Additional Dependencies**' property by clicking the '**...**' button.



The window titled "Additional Dependencies" should open: type ImagetoPDFDLL.lib and click the '**OK**' button. Click '**OK**' to close "Property Pages" window. You have successfully linked Imageto`PDFDLL.lib` with your project.

Using methods from DLL file

After linking with ImagetoPDFDLL.lib file you can use methods contained in ImagertoPDFDLL.dll file. Simply include the ImagetoPDFDLL.h file in all source files that need to use those methods.

Using the Investintech ImagetoPDFDLL Conversion DLL in Microsoft Visual C++ .NET 2003

In this section you will learn how to use the Investintech PDF2Excel Conversion DLL. All pictures and examples from now on will be based on Microsoft Visual C++ .NET 2003.

Investintech ImagetoPDFDLL Conversion DLL Methods

In this section you will find details about each method such as: prototype; description; calling conventions; parameters; return value; and sample use of the methods.

Interface

The interface of the Investintech ImagetoPDF Conversion DLL consists of its exposed methods. The interface contains an exposed method for each combination of: calling convention; parameter type; and error handling.

The calling convention is either the “cdecl” calling convention or the “stdcall” calling convention.

The “cdecl” calling convention is normally used for calling C/C++ functions. It is characterized by the following points:

- arguments are passed in order from right to left
- the calling function is responsible for popping the arguments from the stack at the conclusion of function execution
- an underscore character (_) is prefixed to function names
- no case translation is performed on the function name

The “stdcall” calling convention is typically used to call Windows operating system API functions. It is characterized by the following points:

- arguments are passed in order from right to left
- arguments are passed by value, unless a pointer or reference type is passed.
- The called function is responsible for popping its own arguments from the stack.
- an underscore (_) is prefixed to the name. The name is followed by the at sign (@) followed by the number of bytes (in decimal) in the argument list. For instance, the function declared as void foo(int bar) is decorated as follows: _foo@4
- no case-translation is performed on the function name

Parameter Type

The parameter type is either ANSI (C char *) or UNICODE (BSTR).

File Names

Files may be specified by supplying relative or absolute file names. A relative file name is relative to the application default directory containing the application executable file (e.g. “..\in_parent_dir.pdf”, “in_application_dir.pdf”, “subdir\in_sub_directory.pdf”). An absolute file name includes the directory path to the file beginning at the top of the directory tree (e.g. “c:\pdfs\absolute_file.pdf”).

It is also possible to use UNC notation (e.g. \\myserver\public\file.pdf”) provided your user account has sufficient access permissions.

Error Handling

The error handling strategy is either to throw an exception (part of the C++ throw-catch error handling construct) or return in integer/boolean type result code (the C style error handling strategy).

The ImagetoPDF Conversion DLL identifies the characteristics of a function by an encoded suffix. The suffix is defined by the rules:

- C or S - cdecl or stdcall calling convention
- B or L - BSTR or LPST parameter type
- R or T - return error code or throw exception

Common Sample Source Code

All sample code snippets share the following initialization code snippet:

```
//.h file must be included if we want to use conversion methods
#include <afx.h>
#include <afxwin.h>         // MFC core and standard components
#include "ImagetoPDFDLL.h"

CString str_inputFile = _T("c:\pdfs\sample_image.bmp"); //input
file name
CString str_outputFile = _T(“image.pdf”); //output file name

//conversion from CString to BSTR
BSTR bstr_inputFile = str_inputFile.AllocSysString();
```

```
BSTR bstr_outputFile = str_output.AllocSysString();
```

Conversion from Image file to PDF document

The following methods convert an image file to PDF document file.

Image_to_PDF _CBR

Prototype

```
bool Image_to_PDF _CBR(BSTR inFile, BSTR outFile)
```

Description

The Image_to_PDF_CBR**()** method converts the user specified image file to PDF format. The conversion type is implicitly specified by the image file. The PDF file may be viewed by Sonic PDF Creator. The original image file is not modified.

Calling Convention

The function uses the “cdecl” calling convention.

Parameters

`inFile`	Unicode string containing the absolute or relative filename of the image file that will be converted to PDF format.
`outFile`	Unicode string containing the absolute or relative filename of the directory that will contain the PDF files created by the conversion.

Returns

`true` if the images files are successfully converted to XLS, `false` otherwise.

Example

```
bool ret = Image_to_PDF_CBR(bstr_inputFile, bstr_outputFile);
```

Image_to_PDF_CBT

Prototype

```
bool Image_to_PDF_CBT(BSTR inFile, BSTR outFile)
```

Description

The Image_to_PDF_CBT**()** method converts the user specified image file to PDF format. The conversion type is implicitly specified by the image file. The PDF file may be viewed by Sonic PDF Creator. The original image file is not modified.

Calling Convention

The function uses the “cdecl” calling convention.

Parameters

`inFile`	Unicode string containing the absolute or relative filename of the image file that will be converted to XLS format.
`outFile`	Unicode string containing the absolute or relative filename of the PDF file containing the result of conversion.

Returns

`true` if the image file is successfully converted to PDF format, otherwise throws an exception that should be handled by the calling application.

Example

```
try
{
  bool ret  = Image_to_PDF_ CBT(bstr_inputFile, bstr_outputFile);
}
catch(…)
{
        // TODO: Handle exception
}
```

Image_to_PDF_CLR

Prototype

```
bool Image_to_PDF_CLR(LPSTR inFile, LPSTR outFile)
```

Description

The Image_to_PDF_CLR**()** method converts the user specified image file to PDF format. The conversion type is implicitly specified by the image file. The PDF file may be viewed by Sonic PDF Creator. The original image file is not modified.

Calling Convention

The function uses the “cdecl” calling convention.

Parameters

`inFile`	ANSI string containing the absolute or relative filename of the image file that will be converted to PDF format.
`outFile`	ANSI string containing the absolute or relative filename of the PDF file containing the result of conversion.

Returns

`true` if the image file is successfully converted to PDF, false otherwise.

Example

```
bool ret = Image_to_PDF_CLR(str_inputFile.GetBuffer(), str_outputFile.GetBuffer());
```

Image_to_PDF_CLT

Prototype

```
bool Image_to_PDF_CLT(LPSTR inFile, LPSTR outFile)
```

Description

The Image_to_PDF_CLT**()** method converts the user specified image file to PDF format. The conversion type is implicitly specified by the image file. The PDF file may be viewed by Sonic PDF Creator. The original image file is not modified.

Calling Convention

The function uses the “cdecl” calling convention.

Parameters

`inFile`	ANSI string containing the absolute or relative filename of the image file that will be converted to PDF format.

`outFile` ANSI string containing the absolute or relative filename of the PDF file containing the result of conversion.

Returns

`true` if the image file is successfully converted to PDF, otherwise throws an exception that the calling application should handle.

Example

```
try
{
  bool ret = Image_to_PDF_CLT(str_inputFile.GetBuffer(), str_outputFile.GetBuffer());
}
catch(…)
{
        // TODO: Handle exception
}
```

Image_to_PDF_SBR

Prototype

```
UINT Image_to_PDF_SBR(BSTR inFile, BSTR outFile)
```

Description

The Image_to_PDF_SBR**()** method converts the user specified image file to PDF format. The conversion type is implicitly specified by the image file. The PDF file may be viewed by Sonic PDF Creator. The original image file is not modified.

Calling Convention

The function uses the “stdcall” calling convention.

Parameters

`inFile` Unicode string containing the absolute or relative filename of the image file that will be converted to PDF format.

`outFile` Unicode string containing the absolute or relative filename of the PDF file containing the result of conversion.

Returns

`Non-zero integer value` if the image file is successfully converted to PDF, zero integer value otherwise.

Example

```
UINT ret = Image_to_PDF_SBR(bstr_inputFile, bstr_outputFile);
```

Imahe_to_PDF_SBT

Prototype

```
UINT Image_to_PDF_SBT(BSTR inFile, BSTR outFile)
```

Description

The Image_to_PDF_SBT**()** method converts the user specified image file to PDF format. The conversion type is implicitly specified by the image file. The PDF file may be viewed by Sonic PDF Creator. The original image file is not modified.

Calling Convention

The function uses the “stdcall” calling convention.

Parameters

`inFile`	Unicode string containing the absolute or relative filename of the image file that will be converted to PDF format.
`outFile`	Unicode string containing the absolute or relative filename of the PDF file containing the result of conversion.

Returns

`Non-zero integer value` if the image file is successfully converted to PDF, otherwise throws an exception that the calling application should handle.

Example

```
try
{
  UINT ret = Image_to_PDF_SBT(bstr_inputFile, bstr_outputFile);
}
catch(…)
{
```

```
        // TODO: Handle exception
}
```

Image_to_PDF_SLR

Prototype

```
UINT Image_to_PDF_SLR(LPSTR inFile, LPSTR outFile)
```

Description

The Image_to_PDF_SLR**()** method converts the user specified image file to PDF format. The conversion type is implicitly specified by the image file. The PDF file may be viewed by Sonic PDF Creator. The original image file is not modified.

Calling Convention

The function uses the “stdcall” calling convention.

Parameters

`inFile`	ANSI string containing the absolute or relative filename of the image file that will be converted to PDF format.
`outFile`	ANSI string containing the absolute or relative filename of the PDF file containing the result of conversion.

Returns

`Non-zero integer value` if time image file is successfully converted to PDF, zero integer value otherwise.

Example

```
UINT retCode = Image_to_PDF_SLR(str_inputFile, str_outputFile);
```

Image_to_PDF_SLT

Prototype

```
UINT Image_to_PDF_SLT(LPSTR inFile, LPSTR outFile)
```

Description

The Image_PDF_to_Excel_SLT**()** method converts the user specified image file to PDF format. The conversion type is implicitly specified by the image file. The PDF file may be viewed by Sonic PDF Creator. The original image file is not modified.

Calling Convention

The function uses the “stdcall” calling convention.

Parameters

`inFile`	ANSI string containing absolute or relative filename of the image file that will be converted to PDF format.
`outFile`	ANSI string containing the absolute or relative filename of the PDF file containing the result of conversion.

Returns

`Non-zero integer value` if the image file is successfully converted to PDF, otherwise.throws an exception that should be handled by the calling application.

Example

```
try
{
  UINT retCode = PDF_to_Excel_SLT(str_inputFile, str_outputFile);
}
catch(...)
{
  //TODO: handle error
}
```

VB6 callable code

VB6 client applications must use these functions in order to avoid runtime errors related to stack and string references.

Image_to_PDF_VB6

Prototype

```
UINT Image_to_PDF_VB6(BSTR inFile, BSTR outFile)
```

Description

The Image_to_PDF_VB6**()** method converts the user specified image file to PDF format. The conversion type is implicitly specified by the image file. The PDF file may be viewed by Sonic PDF Creator. The original image file is not modified.

Calling Convention

The function uses the “stdcall” calling convention.

Parameters

`inFile`	Unicode string containing the absolute or relative filename of the image file that will be converted to PDF format.
`outFile`	Unicode string containing the absolute or relative filename of the PDF file containing the result of conversion.

Returns

`Non-zero integer value` if the image file is successfully converted to PDF, zero integer value otherwise.

Example

```
UINT ret = Image_to_PDF_VB6(bstr_inputFile, bstr_outputFile);
```

InvestintechConversionDLL_IDString

```
char* InvestintechConversionDLL_IDString()
```

Description

The **`InvestintechConversionDLL_IDString()`** method returns vendor's ID string.

Parameters

None.

Returns

"Investintech Conversion DLL Version 3.00 by Investintech.com Inc." string.

Investintech ImagetoPDF Conversion COM Server Methods

This section provides a description of the conversion methods exposed by the ImagetoPDF COM Server. The COM Server defines the CoClass CImagetoPDF that implements the interface IImagetoPDF. The interface consists of the following methods.

ImagetoPDF

Prototype

```
HRESULT  ImagetoPDF ([in] BSTR sourceFile, [in] BSTR destFile,
[out,retval] VARIANT_BOOL *successFlag)
```

Description

The ImagetoPDF**()** method converts the user specified image file to PDF format. The conversion type is implicitly specified by the image file. The output file may be opened with Sonic PDF Creator. The original image file is not modified.

Parameters

`sourceFile`	Unicode string containing the absolute or relative filename of the image file that will be converted to PDF format.
`destFile`	Unicode string containing the absolute or relative filename of the PDF file that contains the result of conversion.
`SuccessFlag`	Pointer to boolean variable that will contain the result of the operation: true is the conversion operation is successful; else false.

Returns

`HRESULT enumerated value indicating the result of the COM operation`.

Example

```
HRESULT result = ImagetoPDF(BSTR_sourceFile BSTR_destinationFile, &boolFlag);
```

Index

A
ImagetoPDf conversion Library
Installation Instructions 4
System Requirements 4
C
Conversion DLL 3
Customer Service 2
I
Investintech.com Inc. 1
InvestintechSDK DLL 4
Implicit Linking 11
InvestintechSDK_IDString 45
InvestintechSDK_PDF_to_Excel21 15, 16, 17, 29, 30, 31, 32, 33, 34, 44
Methods 13, 46
InvestintechSDK.dll 11
InvestintechSDK.h 11
InvestintechSDK.lib 11
T
Technical Support 2