

# Foxit PDF SDK 7.5 Upgrade Warnings

This document contains a list of items that developers looking to upgrade to Foxit PDF SDK 7.5 from previous version should pay attention to. Changes to code may be required in order for your projects to compile using the new version.

## Upgrading to version 7.5 from version 7.4

There are two minor upgrade warnings in this release when moving from version 7.4 to version 7.5. In **foxit::pdf::HeaderFooter** Class:

- The parameters of constructor function of **HeaderFooter** have been changed:

SDK 7.4:

```
HeaderFooter(const common::Font& font, float text_size, RGB text_color, const
PageNumberRange& page_range, RectF page_margin,
bool has_text_shrunked, bool has_fixedsize_for_print, int start_page_number,
const HeaderFooterContent& content)
```

SDK 7.5:

```
HeaderFooter(const common::Font& font, float text_size, RGB text_color, const
PageNumberRange& page_range, RectF page_margin,
bool has_text_shrunked, bool has_fixedsize_for_print, int start_page_number,
const HeaderFooterContent& content, bool is_to_embed_font, bool is_underline)
```

- The parameters of **Set** function have been changed:

SDK 7.4:

```
void Set(const common::Font& font, float text_size, RGB text_color, const
PageNumberRange& page_range, RectF page_margin,
bool has_text_shrunked, bool has_fixedsize_for_print, int start_page_number,
const HeaderFooterContent& content)
```

SDK 7.5:

```
void Set(const common::Font& font, float text_size, RGB text_color, const
PageNumberRange& page_range, RectF page_margin,
bool has_text_shrunked, bool has_fixedsize_for_print, int start_page_number,
const HeaderFooterContent& content, bool is_to_embed_font, bool is_underline)
```

## Upgrading to version 7.4 from version 7.3

If you are upgrading from version 7.3 to version 7.4, please take note of the changes that occurred between version 7.3 and version 7.4.

- The value (**e\_ImageCompressjpeg2**) of the enumeration class `foxit::addon::optimization::MonolImageSettings::MonolImageCompressMode` has been renamed to **e\_ImageCompressbig2**.
- In `foxit::pdf::annots::Markup` class:

The functions below have been removed from version 7.4:

```
int GetStateAnnotCount(StateModel model);
Note GetStateAnnot(StateModel model, int index);
```

And then add a new function as below:

```
NoteArray GetStateAnnots(StateModel model);
```

- In `foxit::ActionCallback` class, add three callback functions as below:

```
virtual pdf::PDFDoc GetCurrentDoc() = 0;
virtual pdf::PDFDoc CreateBlankDoc() = 0;
virtual bool OpenDoc(const WString& file_path, const WString& password) = 0;
```

- Some APIs have been changed in version 7.4. The following table lists the APIs mapping between version 7.3 and 7.4.

Class	SDK 7.3	SDK 7.4
foxit::addon::conversion::Word2PDFSettingData (only for Windows platform)	Word2PDFSettingData (bool include_doc_props, <b>bool convert_to_pdfa</b> , ConvertOptimizeOption optimize_option, ConvertContentOption content_option, ConvertBookmarkOption bookmark_option)	Word2PDFSettingData (bool include_doc_props, ConvertOptimizeOption optimize_option, ConvertContentOption content_option, ConvertBookmarkOption bookmark_option, <b>bool convert_to_pdfa</b> )
	void Set (bool include_doc_props, <b>bool convert_to_pdfa</b> , ConvertOptimizeOption optimize_option, ConvertContentOption content_option, ConvertBookmarkOption bookmark_option)	void Set (bool include_doc_props, ConvertOptimizeOption optimize_option, ConvertContentOption content_option, ConvertBookmarkOption bookmark_option, <b>bool convert_to_pdfa</b> )
foxit::addon::conversion::Excel2PDFSettingData (only for Windows platform)	Excel2PDFSettingData (bool include_doc_props, <b>bool convert_to_pdfa</b> , ConvertQuality	Excel2PDFSettingData (bool include_doc_props, ConvertQuality quality, bool ignore_print_area,

Class	SDK 7.3	SDK 7.4
	quality, bool ignore_print_area, ScaleType scale_type) void Set(bool include_doc_props, <b>bool convert_to_pdfa</b> , ConvertQuality quality, bool ignore_print_area, ScaleType scale_type)	ScaleType scale_type, <b>bool convert_to_pdfa</b> ) void Set(bool include_doc_props, ConvertQuality quality, bool ignore_print_area, ScaleType scale_type, <b>bool convert_to_pdfa</b> )
foxit::pdf::annots::Markup	Note AddStateAnnot(StateModel model, State state);	Note AddStateAnnot( <b>const WString&amp; title</b> , StateModel model, State state);
foxit::pdf::TextSearch	explicit TextSearch(const PDFDoc& document, SearchCancelCallback* cancel = NULL);	explicit TextSearch(const PDFDoc& document, SearchCancelCallback* cancel = NULL, <b>int flags = foxit::pdf::TextPage::e_ParseText Normal</b> );
foxit::pdf::annots::FreeText	void SetTextMatrix(const Matrix& text_matrix)	deprecated
	Matrix GetTextMatrix() const	deprecated

## Upgrading to version 7.3 from version 7.2

There are some minor upgrade warnings in this release when moving from version 7.2 to version 7.3:

- The parameter of constructor function of **Font** has been changed, which deletes the default value for "weight".  
SDK 7.2: **Font**(const wchar\_t\* name, uint32 styles, Charset charset, int **weight = 0**);  
SDK 7.3: **Font**(const wchar\_t\* name, uint32 styles, Charset charset, int **weight**);
- The enumeration **foxit::pdf::graphics::ImageObject::ColorSpace** has been changed to **foxit::common::ColorSpaceType**.
- The enumeration **foxit::pdf::interform::Field::Flags::e\_FlagChoiseMultiSelect** has been changed to **foxit::pdf::interform::Field::Type::e\_FlagChoiceMultiSelect**.
- The TrustedCertStoreCallback class adds a new callback function (**virtual bool IsCertTrustedRoot(const String& cert) = 0**).

## Upgrading to version 7.2 from version 7.1

There is only one minor upgrade warning in this release when moving from version 7.1 to version 7.2. In this release we worked hard to ensure that no modifications to existing code bases would be required when upgrading.

The parameter type of the function **foxit::addon::optimization::ImageSettings::SetQuality** has been changed as below:

#### SDK 7.1:

```
void foxit::addon::optimization::ImageSettings::SetQuality (int32 quality)
```

#### SDK 7.2:

```
void foxit::addon::optimization::ImageSettings::SetQuality (ImageCompressQuality quality)
```

### Upgrading to version 7.1 from version 7.0

If you are upgrading from version 7.0 to version 7.1, please take note of the changes that occurred between version 7.0 and version 7.1.

- From 7.1 version, the whole **foxit::addon::ConnectedPDF** class has been removed.
- Some APIs have been changed in version 7.1. Note that the APIs related to ConnectedPDF in the PDFDoc class have also been removed. The following table lists the APIs mapping between version 7.0 and 7.1.

SDK 7.0	SDK 7.1
<b>Charset</b> foxit::common::Font::GetCharset(const pdf::PDFDoc & document)	<b>CIDCharset</b> foxit::common::Font::GetCIDCharset(const pdf::PDFDoc & document)
void foxit::pdf::annots::Line::SetMeasureUnit(int measure_type, const String & unit)	void foxit::pdf::annots::Line::SetMeasureUnit(MeasureType measure_type, const String & unit)
String foxit::pdf::annots::Line::GetMeasureUnit(int measure_type)	String foxit::pdf::annots::Line::GetMeasureUnit(MeasureType measure_type)
void foxit::pdf::annots::Line::SetMeasureConversionFactor(int measure_type, float factor)	void foxit::pdf::annots::Line::SetMeasureConversionFactor(MeasureType measure_type, float factor)
float foxit::pdf::annots::Line::GetMeasureConversionFactor(int measure_type)	float foxit::pdf::annots::Line::GetMeasureConversionFactor(MeasureType measure_type)
bool foxit::pdf::PDFDoc::IsCPDF() const	Removed
bool foxit::pdf::PDFDoc::IsCDRM() const	Removed
String foxit::pdf::PDFDoc::GetDocURI()	Removed
String foxit::pdf::PDFDoc::GetEndpoint()	Removed

### Upgrading to version 7.0 from version 6.4

There are two minor upgrade warnings in this release when moving from version 6.4 to version 7.0:

- The value of enumeration **foxit::pdf::Signature::e\_StateUnknown** has been changed from 0 to 0x80000000.
- The functions **foxit::pdf::Signature::StartVerify** and **foxit::pdf::SignatureCallback::VerifySigState** are only used to verify the integrity of a signature instead of the validity of a signature, so that the return value of **VerifySigState** in the default signature callback has been changed from "VALID/INVALID" to "NOCHANGE/CHANGE".

### Upgrading to version 6.4 from version 6.3

There are some minor upgrade warnings in this release when moving from version 6.3 to version 6.4.

- The functions that contain the parameter "**ifstream**" have been deprecated.
- The function **foxit::common::Renderer::SetTransformAnnotIcon** has been deprecated.
- The parameter "**is\_transform\_icon**" of the function **foxit::pdf::annots::Annot::GetDeviceRect** has been deprecated.

### Upgrading to version 6.3 from version 6.2.1

There is only one minor upgrade warning in this release when moving from version 6.2.1 to version 6.3. In this release we worked hard to ensure that no modifications to existing code bases would be required when upgrading.

The function name of **foxit::pdf::interform::Field::SetDefaultValue** has been changed to **foxit::pdf::interform::Field::SetDefaultVlaue**.

### Upgrading to version 6.2.1 from version 6.2

There is only one upgrade warning in this release when moving from version 6.2 to version 6.2.1. In this release we worked hard to ensure that no modifications to existing code bases would be required when upgrading.

Add a callback interface **WidgetEvent** in **foxit::addon::xfa::DocProviderCallback** class to execute xfa widget event when you create or destroy xfa widget.

## Upgrading to version 6.2 from version 6.1.1

There is only one upgrade warning in this release when moving from version 6.1.1 to version 6.2. In this release we worked hard to ensure that no modifications to existing code bases would be required when upgrading.

The return type of `foxit::addon::xfa::XFADoc::ExportData` has been changed from **void** to **bool**.

## Upgrading to version 6.1.1 from version 6.1

If you are upgrading from version 6.1 to version 6.1.1, please take note of the changes that occurred between version 6.1 and version 6.1.1.

For the version 6.1.1 release of Foxit PDF SDK, there are two minor changes:

- Add a new class "`foxit::common::file::StreamCallback`", and change the parameter in the **SaveAs** function from  
`"bool Image::SaveAs(file::WriterCallback* file, const wchar_t* file_extension)  
EXCE_SPEC"` to  
`"bool Image::SaveAs(file::StreamCallback* file, const wchar_t* file_extension)  
EXCE_SPEC"`.
- Change the enum definitions of watermark position from "**enum  
foxit::pdf::WatermarkSettings::Position**" to "**enum foxit::common::Position**".

If you use any of these functions in your code then you may need to update your code to reflect the change when moving to version 6.1.1.

## Upgrading to version 6.1 from version 6.0

There are no upgrade warnings when moving from version 6.0 to version 6.1. In this release we worked hard to ensure that no modifications to existing code bases would be required when upgrading.

## Upgrading to version 6.0 from version 5.3

Foxit PDF SDK 6.0 is a major new version of Foxit PDF SDK and contains a lot of changes to the APIs. In version 6.0, the APIs have been substantially changed, which are provided from C programming language to C++ programming language. Please take note of the changes that occurred between version 5.3 and version 6.0 if you are upgrading from version 5.3 to version 6.0.

The following table lists the APIs mapping between version 5.3 and 6.0.

5.3 C API		6.0 C++ API	
API	Memo	API	Memo
<b>FSCRT_Annot_GetPage</b>		foxit::pdf::annots::Annot::GetPage() const	
<b>FSCRT_Archive_Create</b>		N/A	
<b>FSCRT_Archive_GetData</b>		N/A	
<b>FSCRT_Archive_LoadData</b>		N/A	
<b>FSCRT_Archive_Release</b>		N/A	
<b>FSCRT_Array_Clear</b>		Not Required.	GSDK 6.0 provides specific type of Array directly.
<b>FSCRT_Array_Init</b>		Not Required.	GSDK 6.0 provides specific type of Array directly.
<b>FSCRT_Barcode_GenerateBitmap</b>		foxit::common::Barcode::GenerateBitmap(const WString& info, Format format, int32 unit_width, int32 unit_height, QRErrorCorrectionLevel level)	
<b>FSCRT_Base64_DecodeToBuffer</b>		foxit::common::Codec::Base64Decode(const void* encoded_data_buffer, size_t length)	
<b>FSCRT_Base64_EncodeFromBuffer</b>		foxit::common::Codec::Base64Encode(const void* data_buffer, size_t length)	
<b>FSCRT_BCModule_Finalize</b>		Not Required.	It will be destroyed automatically with SDK.
<b>FSCRT_BCModule_Initialize</b>		Not Required.	It will be initialized automatically with SDK.
<b>FSCRT_Bitmap_CalcBBox</b>		foxit::common::Bitmap::CalculateBBoxByColor(ARGB backgroud_color)	
<b>FSCRT_Bitmap_Clone</b>		foxit::common::Bitmap::Clone(const RectI* clip_rect = NULL) const	
<b>FSCRT_Bitmap_ConvertFormat</b>		foxit::common::Bitmap::ConvertFormat(DIBFormat format, void* icc_transform = NULL)	
<b>FSCRT_Bitmap_Create</b>		foxit::common::Bitmap::Bitmap(int width, int height, DIBFormat format, uint8* buffer = NULL, int pitch = 0)	
<b>FSCRT_Bitmap_FillRect</b>		foxit::common::Bitmap::FillRect(ARGB color, const RectI* rect = NULL)	
<b>FSCRT_Bitmap_GetBitmapInfo</b>		N/A	
<b>FSCRT_Bitmap_GetFlipped</b>		foxit::common::Bitmap::Flip(bool is_flip_horz, bool is_flip_vert)	
<b>FSCRT_Bitmap_GetFormat</b>		foxit::common::Bitmap::GetFormat() const	
<b>FSCRT_Bitmap_GetLineBuffer</b>	Get the first address of a specific line buffer of a bitmap object.	foxit::common::Bitmap::GetBuffer() const	Get the first address of the bitmap buffer.
<b>FSCRT_Bitmap_GetLineStride</b>		foxit::common::Bitmap::GetPitch() const	
<b>FSCRT_Bitmap_GetMask</b>		foxit::common::Bitmap::GetMask(const RectI* clip_rect = NULL)	
<b>FSCRT_Bitmap_GetSize</b>		foxit::common::Bitmap::GetWidth() const foxit::common::Bitmap::GetHeight() const	
<b>FSCRT_Bitmap_Release</b>		foxit::common::Bitmap::~~Bitmap()	
<b>FSCRT_Bitmap_StretchTo</b>		foxit::common::Bitmap::StretchTo(int dest_width, int dest_height, InterpolationFlag flag, const RectI* clip_rect = NULL)	

5.3 C API		6.0 C++ API	
<b>FSCRT_Bitmap_TransformTo</b>		foxit::common::Bitmap::TransformTo(const Matrix& matrix, InterpolationFlag flag, int& out_left, int& out_top, const RectI* clip_rect = NULL)	
<b>FSCRT_BStr_Clear</b>		Replaced by class <b>String</b> and class <b>WString</b>	
<b>FSCRT_BStr_Init</b>		Replaced by class <b>String</b> and class <b>WString</b>	
<b>FSCRT_BStr_Set</b>		Replaced by class <b>String</b> and class <b>WString</b>	
<b>FSCRT_BStr_SetLength</b>		Replaced by class <b>String</b> and class <b>WString</b>	
<b>FSCRT_Digest_Finish</b>		N/A	
<b>FSCRT_Digest_Start</b>		N/A	
<b>FSCRT_Digest_Update</b>		N/A	
<b>FSCRT_Doc_GetType</b>	Get a document type (fdf or pdf)	// Get a document type (fdf or xfdf) Foxit::fdf::Doc::GetType() const	Fdf, pdf, and xfa are separate classes, so that do not need a specific API to get the type of a document.
<b>FSCRT_File_ClearRange</b>		Not Required.	
<b>FSCRT_File_Create</b>		Not Required.	
<b>FSCRT_File_Create_L</b>		Not Required.	
<b>FSCRT_File_CreateCacheFile</b>		Not Required.	
<b>FSCRT_File_CreateFromFileName</b>		Not Required.	
<b>FSCRT_File_CreateFromMemory</b>		Not Required.	
<b>FSCRT_File_GetSize</b>		Not Required.	
<b>FSCRT_File_Read</b>		Not Required.	
<b>FSCRT_File_Release</b>		Not Required.	
<b>FSCRT_File_SetRange</b>		Not Required.	
<b>FSCRT_File_Write</b>		Not Required.	
<b>FSCRT_Flate_Compress</b>		foxit::common::Codec::FlateCompress(const void* data_buffer, size_t length)	
<b>FSCRT_Flate-Decompress</b>		foxit::common::Codec::FlateDecompress(const void* compressed_data_buffer, size_t length)	
<b>FSCRT_Font_Create</b>		foxit::common::Font::Font(const wchar_t* name, uint32 styles, CharSet charset, int weight = 0)	
<b>FSCRT_Font_CreateFromFile</b>		foxit::common::Font::Font(const char* font_file_path, int face_index, CharSet charset) foxit::common::Font::Font(const wchar_t* font_file_path, int face_index, CharSet charset)	
<b>FSCRT_Font_CreateStandard</b>		foxit::common::Font::Font(StandardID font_id)	
<b>FSCRT_Font_EnumFaces</b>		N/A	
<b>FSCRT_Font_GetAscent</b>		foxit::common::Font::GetAscent() const	
<b>FSCRT_Font_GetCharBBox</b>		foxit::common::Font::GetCharBBox(uint32 unicode) const	
<b>FSCRT_Font_GetCharWidth</b>		foxit::common::Font::GetCharWidth(uint32 unicode) const	
<b>FSCRT_Font_GetDescent</b>		foxit::common::Font::GetDescent() const	
<b>FSCRT_Font_GetName</b>		foxit::common::Font::GetName()	
<b>FSCRT_Font_IsBold</b>		foxit::common::Font::IsBold() const	
<b>FSCRT_Font_IsItalic</b>		foxit::common::Font::IsItalic() const	
<b>FSCRT_Font_Release</b>		foxit::common::Font::~Font()	
<b>FSCRT_Image_CountFrames</b>		foxit::common::Image::GetFrameCount() const	



5.3 C API		6.0 C++ API	
<b>FSCRT_Image_GetCurrentFrameBitmap</b>		foxit::common::Image::GetFrameBitmap(int index) const	Get frame bitmap through index
<b>FSCRT_Image_GetCurrentFrameSize</b>		foxit::common::Image::GetFrameBitmap(int index) const foxit::common::Bitmap::GetWidth() const foxit::common::Bitmap::GetHeight() const	Get frame bitmap through index
<b>FSCRT_Image_GetProperty</b>		foxit::common::Image::GetX DPI() const foxit::common::Image::GetY DPI() const	
<b>FSCRT_Image_GetSize</b>		foxit::common::Image::GetWidth() const foxit::common::Image::GetHeight() const	
<b>FSCRT_Image_GetType</b>		foxit::common::Image::GetType() const	
<b>FSCRT_Image_LoadFrame</b>		Not Required.	Get bitmap through index.
<b>FSCRT_Image_LoadFromFile</b>		foxit::common::Image::Image(const char* path) foxit::common::Image::Image(const wchar_t* path) foxit::common::Image::Image(const void* buffer, size_t length) foxit::common::Image::Image(foxit::common::file::ReaderCallback* file_read) foxit::common::Image::Image(std::ifstream* input_file_stream)	
<b>FSCRT_Image_Release</b>		foxit::common::Image::~Image()	
<b>FSCRT_ImageFile_AddFrame</b>		foxit::common::Image::AddFrame(const Bitmap& bitmap)	
<b>FSCRT_ImageFile_Create</b>		foxit::common::Image::Image()	
<b>FSCRT_ImageFile_Release</b>		foxit::common::Image::~Image()	
<b>FSCRT_ImageFile_SetProperty</b>		foxit::common::Image::SetDPIS(int x_dpi, int y_dpi)	
<b>FSCRT_Library_AddFontFile</b>		N/A	
<b>FSCRT_Library_CreateDefaultMgr</b>		foxit::common::Library::Initialize(const char* sn, const char* key)	
<b>FSCRT_Library_CreateMgr</b>		N/A	
<b>FSCRT_Library_DestroyMgr</b>		foxit::common::Library::Release()	
<b>FSCRT_Library_EnableJavaScript</b>		foxit::common::Library::EnableJavaScript(bool enable_javascript)	
<b>FSCRT_Library_GetFixedMemSize</b>		N/A	
<b>FSCRT_Library_GetMgr</b>		N/A	
<b>FSCRT_Library_GetPrivateData</b>		N/A	
<b>FSCRT_Library_GetVersion</b>		foxit::common::Library::GetVersion()	
<b>FSCRT_Library_LoadSystemFonts</b>		N/A	
<b>FSCRT_Library_OOMRecover</b>		Not Required.	
<b>FSCRT_Library_RemovePrivateData</b>		N/A	
<b>FSCRT_Library_SetAppHandler</b>		N/A	
<b>FSCRT_Library_SetCacheSize</b>		foxit::common::Library::SetCacheSize(uint32 size)	
<b>FSCRT_Library_SetFontMapperHandler</b>		N/A	
<b>FSCRT_Library_SetLogFile</b>		foxit::common::Library::SetLogFile(const char* log_file_path) foxit::common::Library::SetLogFile(const wchar_t* log_file_path)	
<b>FSCRT_Library_SetMgr</b>		N/A	
<b>FSCRT_Library_SetPrivateData</b>		N/A	
<b>FSCRT_Library_SetThreadHandler</b>		N/A	
<b>FSCRT_Library_TriggerRecover</b>		Not Required.	
<b>FSCRT_License_GetType</b>		N/A	

5.3 C API		6.0 C++ API	
<b>FSCRT_License_UnlockLibrary</b>		foxit::common::Library::Initialize(const char* sn, const char* key)	
<b>FSCRT_Matrix_Concat</b>		foxit::Matrix::Concat	
<b>FSCRT_Matrix_GetReverse</b>		foxit::Matrix dst_matrix; dst_matrix.SetReverse(src_matrix);	
<b>FSCRT_Matrix_Rotate</b>		foxit::Matrix::Rotate	
<b>FSCRT_Matrix_Scale</b>		foxit::Matrix::Scale	
<b>FSCRT_Matrix_Shear</b>		foxit::Matrix::Shear	
<b>FSCRT_Matrix_TransformPoint</b>		foxit::Matrix::TransformPoint	
<b>FSCRT_Matrix_TransformPointF</b>		foxit::Matrix::TransformPoint	
<b>FSCRT_Matrix_TransformRect</b>		foxit::Matrix::TransformRect	
<b>FSCRT_Matrix_TransformRectF</b>		foxit::Matrix::TransformRect	
<b>FSCRT_Matrix_Translate</b>		foxit::Matrix::Translate	
<b>FSCRT_Memory_Alloc</b>		foxit::common::Library::Alloc(size_t size)	
<b>FSCRT_Memory_Free</b>		foxit::common::Library::Free(void* ptr)	
<b>FSCRT_Memory_Realloc</b>		foxit::common::Library::Realloc(void* ptr, size_t new_size)	
<b>FSCRT_Page_GetDocument</b>		foxit::pdf::Page::GetDocument() const	
<b>FSCRT_PathData_AddPointsCount</b>		Not Required.	
<b>FSCRT_PathData_AppendEllipse</b>		foxit::common::Path::AppendEllipse(const RectF& rect)	
<b>FSCRT_PathData_AppendRect</b>		foxit::common::Path::AppendRect(const RectF& rect)	
<b>FSCRT_PathData_Clear</b>		foxit::common::Path::Clear()	
<b>FSCRT_PathData_CloseFigure</b>		foxit::common::Path::CloseFigure()	
<b>FSCRT_PathData_CountPoints</b>		foxit::common::Path::GetPointCount()	
<b>FSCRT_PathData_Create</b>		foxit::common::Path::Path()	
<b>FSCRT_PathData_CubicBezierTo</b>		foxit::common::Path::CubicBezierTo(const PointF& point1, const PointF& point2, const PointF& point3)	
<b>FSCRT_PathData_GetPoint</b>		foxit::common::Path::GetPoint(int index) foxit::common::Path::GetPointType(int index)	
<b>FSCRT_PathData_LineTo</b>		foxit::common::Path::LineTo(const PointF& point)	
<b>FSCRT_PathData_MoveTo</b>		foxit::common::Path::MoveTo(const PointF& point)	
<b>FSCRT_PathData_Release</b>		foxit::common::Path::~~Path()	
<b>FSCRT_PathData_RemovePoint</b>		foxit::common::Path::RemovePoint(int index)	
<b>FSCRT_PathData_SetPoint</b>		foxit::common::Path::SetPoint(int index, const PointF& point, PointType type)	
<b>FSCRT_PathData_Transform</b>		foxit::common::Path::Transform(const Matrix& matrix)	
<b>FSCRT_PDFModule_Finalize</b>		Not Required.	It will be destroyed automatically with SDK.
<b>FSCRT_PDFModule_Initialize</b>		Not Required.	It will be initialized automatically with SDK.
<b>FSCRT_Progress_Continue</b>		foxit::common::Progressive::Continue()	
<b>FSCRT_Progress_GetPercent</b>		foxit::common::Progressive::GetRateOfProgress()	
<b>FSCRT_Progress_Release</b>		foxit::common::Progressive::~~Progressive()	
<b>FSCRT_Progress_SetProperty</b>		foxit::pdf::Doc::SetFileVersion(int version)	
<b>FSCRT_PSI_AddPoint</b>		foxit::pdf::PSI::AddPoint(const PointF& point, common::Path::PointType type, float pressure)	
<b>FSCRT_PSI_ConvertToPDFAnnot</b>		foxit::pdf::PSI::ConvertToPDFAnnot(const Page& page, const RectF& annot_rect, common::Rotation rotate)	
<b>FSCRT_PSI_Create</b>		foxit::pdf::PSI::PSI(int width, int height, bool simulate)	

5.3 C API		6.0 C++ API	
<b>FSCRT_PSI_GetContentsRect</b>		foxit::pdf::PSI::GetContentsRect()	
<b>FSCRT_PSI_InitCanvas</b>		foxit::pdf::PSI::PSI(int width, int height, bool simulate)	
<b>FSCRT_PSI_Release</b>		foxit::pdf::PSI::~~PSI()	
<b>FSCRT_PSI_Render</b>		foxit::pdf::PSI::GetBitmap() foxit::common::Renderer::StartRenderBitmap(const Bitmap& bitmap, const Matrix& matrix, const RectI* clip_rect = NULL, uint32 interpolation = 0, PauseCallback* pause = NULL)	
<b>FSCRT_PSI_SetInkColor</b>		foxit::pdf::PSI::SetColor(RGB color)	
<b>FSCRT_PSI_SetInkDiameter</b>		foxit::pdf::PSI::SetDiameter(int diameter)	
<b>FSCRT_PSI_SetOpacity</b>		foxit::pdf::PSI::SetOpacity(float opacity)	
<b>FSCRT_Rect_IsAdjacent</b>		N/A	
<b>FSCRT_Renderer_CreateOnBitmap</b>		FSCRT_Renderer_CreateOnBitmap	
<b>FSCRT_Renderer_CreateOnWindowsDC</b>	For Windows Only	foxit::common::Renderer::Renderer(HDC dc)	For Windows Only
<b>FSCRT_Renderer_CreateOnCGContext</b>	For Mac Only	foxit::common::Renderer::Renderer(const CGContextRef& context, DeviceType device_type)	For Mac/iOS Only
<b>FSCRT_Renderer_DrawBitmap</b>		foxit::common::Renderer::StartRenderBitmap(const Bitmap& bitmap, const Matrix& matrix, const RectI* clip_rect = NULL, uint32 interpolation = 0, PauseCallback* pause = NULL)	
<b>FSCRT_Renderer_Release</b>		foxit::common::Renderer::~~Renderer()	
<b>FSCRT_Renderer_SetClipRect</b>		foxit::common::Renderer::SetClipRect(const RectI* clip_rect)	
<b>FSCRT_Renderer_SetDitherBits</b>		N/A	
<b>FSCRT_Renderer_SetFlags</b>		foxit::common::Renderer::SetTransformAnnotIcon(bool is_to_transform_annot_icon) foxit::common::Renderer::SetClearType(bool is_clear_type) foxit::common::Renderer::SetPrintTextAsGraphic(bool is_to_print_text_graphic) foxit::common::Renderer::SetPrintTextAsImage(bool is_to_print_text_image) foxit::common::Renderer::SetForceDownSample(bool is_to_force_down_sample) foxit::common::Renderer::SetRenderPathThinLine(bool is_render_path_thin_line) foxit::common::Renderer::SetRenderTextAntiAliasing(bool is_render_text_antialiasing) foxit::common::Renderer::SetRenderPathAntiAliasing(bool is_render_path_antialiasing) foxit::common::Renderer::SetRenderImageAntiAliasing(bool is_render_image_antialiasing)	
<b>FSCRT_Renderer_SetGamma</b>		foxit::common::Library::SetRenderTextGamma(float gamma)	
<b>FSCRT_UTF8_CountChars</b>		Replaced by class <b>String</b> and class <b>WString</b>	
<b>FSCRT_UTF8_FromUTF16BE</b>		Replaced by class <b>String</b> and class <b>WString</b>	
<b>FSCRT_UTF8_FromUTF16LE</b>		Replaced by class <b>String</b> and class <b>WString</b>	
<b>FSCRT_UTF8_FromUTF32BE</b>		Replaced by class <b>String</b> and class <b>WString</b>	
<b>FSCRT_UTF8_FromUTF32LE</b>		Replaced by class <b>String</b> and class <b>WString</b>	
<b>FSCRT_UTF8_ToUTF16BE</b>		Replaced by class <b>String</b> and class <b>WString</b>	
<b>FSCRT_UTF8_ToUTF16LE</b>		Replaced by class <b>String</b> and class <b>WString</b>	
<b>FSCRT_UTF8_ToUTF32BE</b>		Replaced by class <b>String</b> and class <b>WString</b>	
<b>FSCRT_UTF8_ToUTF32LE</b>		Replaced by class <b>String</b> and class <b>WString</b>	
<b>FSFDF_Annot_ExportToPDFPage</b>		foxit::pdf::Doc::ImportFromFDF(const fdf::Doc& fdf_doc, int types = e_Forms   e_Annots, const common::Range& page_range = common::Range())	
<b>FSFDF_Annot_Get</b>		Not Required.	
<b>FSFDF_Annot_GetCount</b>		Not Required.	

5.3 C API		6.0 C++ API	
<b>FSFDF_Annot_GetPageIndex</b>		Not Required.	
<b>FSFDF_Doc_Close</b>		foxit::fdf::Doc::~Doc()	
<b>FSFDF_Doc_Create</b>		foxit::fdf::Doc::Doc(Type type)	
<b>FSFDF_Doc_ExportAnnotsToPDFDoc</b>		foxit::pdf::Doc::ImportFromFDF(const fdf::Doc& fdf_doc, int types = e_Forms   e_Annots, const common::Range& page_range = common::Range())	
<b>FSFDF_Doc_GetPDFPath</b>		foxit::fdf::Doc::GetPDFPath()	
<b>FSFDF_Doc_Load</b>		foxit::fdf::Doc::Doc(const char* path) foxit::fdf::Doc::Doc(const wchar_t* path) foxit::fdf::Doc::Doc(const void* buffer, size_t length) foxit::fdf::Doc::Doc(common::file::ReaderCallback* file_read) foxit::fdf::Doc::Doc(std::ifstream* input_file_stream)	
<b>FSFDF_Doc_Save</b>		foxit::fdf::Doc::SaveAs(const char* file_path) foxit::fdf::Doc::SaveAs(const wchar_t* file_path) foxit::fdf::Doc::SaveAs(common::file::WriterCallback* file)	
<b>FSFDF_Doc_SetPDFPath</b>		foxit::fdf::Doc::SetPDFPath(const wchar_t* pdf_path)	
<b>FSPDF_ActionData_Clear</b>		Provide class foxit::pdf::actions::Action and its subclass to present action	
<b>FSPDF_ActionData_Init</b>		Provide class foxit::pdf::actions::Action and its subclass to present action	
<b>FSPDF_Annot_Add</b>		foxit::pdf::Page::AddAnnot(annots::Annot::Type type, const RectF& rect)	
<b>FSPDF_Annot_AddState</b>		foxit::pdf::annots::Markup::AddStateAnnot(StateModel model, State state)	
<b>FSPDF_Annot_CountActions</b>		For link annot: // Get link action: foxit::pdf::annots::Link::GetAction() // Count the sub-action of a specific action. foxit::pdf::actions::Action::GetSubActionCount(int index)  For widget annot: // Get widget action: foxit::pdf::annots::Widget::GetAction() or foxit::pdf::actions::AdditionalAction::AdditionalAction(const annots::Annot& annot) foxit::pdf::actions::AdditionalAction::GetAction(TriggerEvent trigger) // Count the sub-action of a specific action. foxit::pdf::actions::Action::GetSubActionCount(int index)	Every action may have sub-action, users can count them by yourself.
<b>FSPDF_Annot_CountReplies</b>		foxit::pdf::annots::Markup::GetReplyCount()	
<b>FSPDF_Annot_CountStateAnnots</b>		foxit::pdf::annots::Markup::GetStateAnnotCount(StateModel model)	
<b>FSPDF_Annot_ExportToFDFDoc</b>		foxit::pdf::Doc::ExportToFDF(const fdf::Doc& fdf_doc, int types = e_Forms   e_Annots, const common::Range& page_range = common::Range())	
<b>FSPDF_Annot_Get</b>		foxit::pdf::Page::GetAnnot(int index)	

5.3 C API		6.0 C++ API	
<b>FSPDF_Annot_GetAction</b>		For link annot: // Get link action: foxit::pdf::annots::Link::GetAction() // Get sub-action from action if sub-action exists foxit::pdf::actions::Action::GetSubAction(int index)  For widget annot: // Get widget action: foxit::pdf::annots::Widget::GetAction() or foxit::pdf::actions::AdditionalAction::AdditionalAction(const annots::Annot& annot) foxit::pdf::actions::AdditionalAction::GetAction(TriggerEvent trigger) // Get sub-action from action if sub-action exists foxit::pdf::actions::Action::GetSubAction(int index)	
<b>FSPDF_Annot_GetAlignment</b>		foxit::pdf::annots::FreeText::GetAlignment() const	
<b>FSPDF_Annot_GetAllByTabOrder</b>		Replaced by class foxit::pdf::TabOrderMgr	
<b>FSPDF_Annot_GetApplyFillColor</b>		foxit::pdf::annots::Redact::GetApplyFillColor() const	
<b>FSPDF_Annot_GetAtDevicePos</b>		foxit::pdf::Page::GetAnnotAtDevicePoint(const PointF& position, float tolerance, const Matrix* matrix = NULL)	
<b>FSPDF_Annot_GetAtPos</b>		foxit::pdf::Page::GetAnnotAtPoint(const PointF& position, float tolerance)	
<b>FSPDF_Annot_GetAttachment</b>		foxit::pdf::annots::FileAttachment::GetFileSpec()	
<b>FSPDF_Annot_GetBorder</b>		foxit::pdf::annots::Annot::GetBorderInfo()	
<b>FSPDF_Annot_GetCalloutLinePoints</b>		foxit::pdf::annots::FreeText::GetCalloutLinePoints()	
<b>FSPDF_Annot_GetCaptionOffset</b>		foxit::pdf::annots::Line::GetCaptionOffset() const	
<b>FSPDF_Annot_GetCaptionPosition</b>		foxit::pdf::annots::Line::GetCaptionPositionType()	
<b>FSPDF_Annot_GetColor</b>		foxit::pdf::annots::Annot::GetBorderColor() foxit::pdf::annots::Square::GetFillColor() const foxit::pdf::annots::Circle::GetFillColor() const foxit::pdf::annots::FreeText::GetFillColor() const foxit::pdf::annots::Polygon::GetFillColor() const foxit::pdf::annots::Redact::GetFillColor() const	
<b>FSPDF_Annot_GetContents</b>		foxit::pdf::annots::Annot::GetContent() const	
<b>FSPDF_Annot_GetCount</b>		foxit::pdf::Page::GetAnnotCount()	
<b>FSPDF_Annot_GetCreationDateTime</b>		foxit::pdf::annots::Markup::GetCreationDateTime() const	
<b>FSPDF_Annot_GetDefaultAppearance</b>		foxit::pdf::annots::FreeText::GetDefaultAppearance()	
<b>FSPDF_Annot_GetDeviceRect</b>		foxit::pdf::annots::Annot::GetDeviceRect(bool is_transform_icon, const Matrix& matrix)	
<b>FSPDF_Annot_GetDict</b>		foxit::pdf::annots::Annot::GetDict()	
<b>FSPDF_Annot_GetFlags</b>		foxit::pdf::annots::Annot::GetFlags() const	
<b>FSPDF_Annot_GetFormControl</b>		foxit::pdf::annots::Widget::GetControl()	
<b>FSPDF_Annot_GetGroupElements</b>		foxit::pdf::annots::Markup::GetGroupElements()	
<b>FSPDF_Annot_GetGroupHeader</b>		foxit::pdf::annots::Markup::GetGroupHeader()	

5.3 C API		6.0 C++ API	
<b>FSPDF_Annot_GetHighlightingMode</b>		foxit::pdf::annots::Link::GetHighlightingMode() foxit::pdf::annots::Widget::GetHighlightingMode()	
<b>FSPDF_Annot_GetIconFit</b>		foxit::pdf::annots::Widget::GetMKIconFit() const	
<b>FSPDF_Annot_GetIconName</b>		foxit::pdf::annots::Note::GetIconName() const foxit::pdf::annots::Stamp::GetIconName() const foxit::pdf::annots::FileAttachment::GetIconName() const	
<b>FSPDF_Annot_GetIndex</b>		foxit::pdf::annots::Annot::GetIndex() const	
<b>FSPDF_Annot_GetInkList</b>		foxit::pdf::annots::Ink::GetInkList()	
<b>FSPDF_Annot_GetInnerRectMargin</b>		foxit::pdf::annots::Caret::GetInnerRect() const foxit::pdf::annots::Circle::GetInnerRect() const foxit::pdf::annots::Square::GetInnerRect() const foxit::pdf::annots::FreeText::GetInnerRect() const	Provide direct function to get inner rect.
<b>FSPDF_Annot_GetIntent</b>		foxit::pdf::annots::Markup::GetIntent() const	
<b>FSPDF_Annot_GetLeaderLineExtension</b>		foxit::pdf::annots::Line::GetLeaderLineExtensionLength() const	
<b>FSPDF_Annot_GetLeaderLineLength</b>		foxit::pdf::annots::Line::GetLeaderLineLength()	
<b>FSPDF_Annot_GetLeaderLineOffset</b>		foxit::pdf::annots::Line::GetLeaderLineOffset() const	
<b>FSPDF_Annot_GetLineEndingStyles</b>		foxit::pdf::annots::Line::GetLineStartStyle() const foxit::pdf::annots::Line::GetLineEndStyle() const foxit::pdf::annots::PolyLine::GetLineStartStyle() const foxit::pdf::annots::PolyLine::GetLineEndStyle() const foxit::pdf::annots::FreeText::GetCalloutLineEndingStyle() const	
<b>FSPDF_Annot_GetLinePosition</b>		foxit::pdf::annots::Line::GetStartPoint() foxit::pdf::annots::Line::GetEndPoint()	
<b>FSPDF_Annot_GetMKCaption</b>		foxit::pdf::annots::Widget::GetMKNormalCaption() const foxit::pdf::annots::Widget::GetMKRolloverCaption() const foxit::pdf::annots::Widget::GetMKDownCaption() const	
<b>FSPDF_Annot_GetMKColor</b>		foxit::pdf::annots::Widget::GetMKBorderColor() const foxit::pdf::annots::Widget::GetMKBackgroundColor() const	
<b>FSPDF_Annot_GetMKIconBitmap</b>		foxit::pdf::annots::Widget::GetMKNormalIconBitmap() foxit::pdf::annots::Widget::GetMKRolloverIconBitmap() foxit::pdf::annots::Widget::GetMKDownIconBitmap()	
<b>FSPDF_Annot_GetMKRotation</b>		foxit::pdf::annots::Widget::GetMKRotation() const	
<b>FSPDF_Annot_GetModifiedDateTime</b>		foxit::pdf::annots::Annot::GetModifiedDateTime() const	
<b>FSPDF_Annot_GetName</b>		foxit::pdf::annots::Annot::GetUniqueID() const	
<b>FSPDF_Annot_GetOpacity</b>		foxit::pdf::annots::Markup::GetOpacity() const foxit::pdf::annots::Screen::GetOpacity() const	
<b>FSPDF_Annot_GetOpenStatus</b>		foxit::pdf::annots::Note::GetOpenStatus() const foxit::pdf::annots::Popup::GetOpenStatus() const	
<b>FSPDF_Annot_GetPopup</b>		foxit::pdf::annots::Markup::GetPopup()	
<b>FSPDF_Annot_GetQuadPoints</b>		foxit::pdf::annots::TextMarkup::GetQuadPoints() const foxit::pdf::annots::Link::GetQuadPoints() const	
<b>FSPDF_Annot_GetRect</b>		foxit::pdf::annots::Annot::GetRect() const	

5.3 C API		6.0 C++ API	
<b>FSPDF_Annot_GetReply</b>		foxit::pdf::annots::Markup::GetReply(int index) const	
<b>FSPDF_Annot_GetState</b>		foxit::pdf::annots::Note::IsStateAnnot()为 true 时, Foxit::pdf::annots::Note::GetStateModel() foxit::pdf::annots::Note::GetState()	
<b>FSPDF_Annot_GetStateAnnot</b>		foxit::pdf::annots::Markup::GetStateAnnot(StateModel model, int index)	
<b>FSPDF_Annot_GetSubject</b>		foxit::pdf::annots::Markup::GetSubject()	
<b>FSPDF_Annot_GetSymbol</b>		N/A	
<b>FSPDF_Annot_GetTitle</b>		foxit::pdf::annots::Markup::GetTitle() const	
<b>FSPDF_Annot_GetType</b>		foxit::pdf::annots::Annot::GetType() const	
<b>FSPDF_Annot_GetVertices</b>		foxit::pdf::annots::Polygon::GetVertexes() foxit::pdf::annots::PolyLine::GetVertexes()	
<b>FSPDF_Annot_HasCaptionContents</b>		foxit::pdf::annots::Line::HasCaption() const	
<b>FSPDF_Annot_HasMKEntry</b>		foxit::pdf::annots::Widget::HasMKEntry(MKEntry mk_entry)	
<b>FSPDF_Annot_InsertAction</b>		For link annot: // Get link action: foxit::pdf::annots::Link::GetAction() // Insert a sub-action foxit::pdf::actions::Action::InsertSubAction(int index, const Action& sub_action)  For widget annot: // Get widget action: foxit::pdf::annots::Widget::GetAction() or foxit::pdf::actions::AdditionalAction::AdditionalAction(const annots::Annot& annot) foxit::pdf::actions::AdditionalAction::GetAction(TriggerEvent trigger) // Insert a sub-action foxit::pdf::actions::Action::InsertSubAction(int index, const Action& sub_action)	
<b>FSPDF_Annot_InsertReply</b>	Only support inserting the reply to the last	foxit::pdf::annots::Markup::AddReply()	
<b>FSPDF_Annot_IsGrouped</b>		foxit::pdf::annots::Markup::IsGrouped()	
<b>FSPDF_Annot_Move</b>		foxit::pdf::annots::Markup::IsGrouped()	
<b>FSPDF_Annot_MoveToFirst</b>		foxit::pdf::Page::MoveAnnotToFirst(const annots::Annot& annot)	
<b>FSPDF_Annot_MoveToLast</b>		foxit::pdf::Page::MoveAnnotToLast(const annots::Annot& annot)	
<b>FSPDF_Annot_MoveToNext</b>		foxit::pdf::Page::MoveAnnotToNext(const annots::Annot& annot)	
<b>FSPDF_Annot_MoveToPrev</b>		foxit::pdf::Page::MoveAnnotToPrev(const annots::Annot& annot)	
<b>FSPDF_Annot_Remove</b>		foxit::pdf::Page::RemoveAnnot(const annots::Annot& annot)	

5.3 C API		6.0 C++ API	
<b>FSPDF_Anot_RemoveAction</b>		For link annot: // Get link action: foxit::pdf::annots::Link::GetAction() // Remove a specified sub-action foxit::pdf::actions::Action::RemoveSubAction(int index) // Remove directly related action foxit::pdf::annots::Link::RemoveAction()  For widget annot: // Get widget action: foxit::pdf::annots::Widget::GetAction() or foxit::pdf::actions::AdditionalAction::AdditionalAction(const annots::Annot& annot) foxit::pdf::actions::AdditionalAction::GetAction(TriggerEvent trigger) // Remove a specified sub-action foxit::pdf::actions::Action::RemoveSubAction(int index) // Remove directly related action foxit::pdf::annots::Widget::RemoveAction() foxit::pdf::actions::AdditionalAction::RemoveAction(TriggerEvent trigger)	
<b>FSPDF_Anot_RemoveAllActions</b>		For link annot: foxit::pdf::annots::Link::RemoveAction()  For widget annot: foxit::pdf::annots::Widget::RemoveAction() or foxit::pdf::actions::AdditionalAction::AdditionalAction(const annots::Annot& annot) foxit::pdf::actions::AdditionalAction::RemoveAllActions()	
<b>FSPDF_Anot_RemoveAllReplies</b>		foxit::pdf::annots::Markup::RemoveAllReplies()	
<b>FSPDF_Anot_RemoveAllStates</b>		foxit::pdf::annots::Markup::RemoveAllStateAnnots()	
<b>FSPDF_Anot_RemoveMKEntry</b>		foxit::pdf::annots::Widget::RemoveMKEntry(MKEntry mk_entry)	
<b>FSPDF_Anot_RemoveReply</b>		foxit::pdf::annots::Markup::RemoveReply(int index)	
<b>FSPDF_Anot_ResetAppearance</b>		foxit::pdf::annots::Annot::ResetAppearanceStream()	
<b>FSPDF_Anot_SearchInAppearanc e</b>		foxit::pdf::TextSearch::TextSearch(const annots::Annot& annot) Then call the functions in class foxit::pdf::TextSearch	



5.3 C API		6.0 C++ API	
<b>FSPDF_Annot_SetAction</b>		For link annot: // Get link action: foxit::pdf::annots::Link::GetAction() // Set a specified sub-action if sub-action exists foxit::pdf::actions::Action::SetSubAction(int index, const Action& sub_action)  // or set directly related action foxit::pdf::annots::Link::SetAction(const actions::Action& action)  For widget annot: // Get widget annot foxit::pdf::annots::Widget::GetAction() or foxit::pdf::actions::AdditionalAction::AdditionalAction(const annots::Annot& annot) foxit::pdf::actions::AdditionalAction::GetAction(TriggerEvent trigger) // Set a specified sub-action if sub-action exists foxit::pdf::actions::Action::SetSubAction(int index, const Action& sub_action)  // or set directly related action foxit::pdf::annots::Widget::SetAction(const actions::Action& action) foxit::pdf::actions::AdditionalAction::SetAction(TriggerEvent trigger, const Action& action)	
<b>FSPDF_Annot_SetAlignment</b>		foxit::pdf::annots::FreeText::SetAlignment(common::Alignment alignment)	
<b>FSPDF_Annot_SetApplyFillColor</b>		foxit::pdf::actions::annots::Redact::SetApplyFillColor(RGB fill_color)	
<b>FSPDF_Annot_SetAttachment</b>		foxit::pdf::annots::FileAttachment::SetFileSpec(const FileSpec& file_spec)	
<b>FSPDF_Annot_SetBorder</b>		foxit::pdf::annots::Annot::SetBorderInfo(const BorderInfo& border)	
<b>FSPDF_Annot_SetCalloutLinePoints</b>		foxit::pdf::annots::FreeText::SetCalloutLinePoints(const PointFArray& point_array)	
<b>FSPDF_Annot_SetCaptionContents</b>		foxit::pdf::annots::Line::EnableCaption(bool cap)	
<b>FSPDF_Annot_SetCaptionOffset</b>		foxit::pdf::annots::Line::SetCaptionOffset(const Offset& offset)	
<b>FSPDF_Annot_SetCaptionPosition</b>		foxit::pdf::annots::Line::SetCaptionPositionType(CapPos cap_position)	
<b>FSPDF_Annot_SetColor</b>		foxit::pdf::annots::Annot::SetBorderColor(RGB color) foxit::pdf::annots::Square::SetFillColor(RGB fill_color) foxit::pdf::annots::Circle::SetFillColor(RGB fill_color) foxit::pdf::annots::FreeText::SetFillColor(RGB fill_color) foxit::pdf::annots::Polygon::SetFillColor(RGB fill_color) foxit::pdf::annots::Redact::SetFillColor(RGB fill_color)	
<b>FSPDF_Annot_SetContents</b>		foxit::pdf::annots::Annot::SetContent(const WString& content)	
<b>FSPDF_Annot_SetCreationDateTime</b>		foxit::pdf::annots::Markup::SetCreationDateTime(const DateTime& date_time)	
<b>FSPDF_Annot_SetDefaultAppearance</b>		foxit::pdf::annots::FreeText::SetDefaultAppearance(const DefaultAppearance& default_ap)	
<b>FSPDF_Annot_SetFlags</b>		foxit::pdf::annots::Annot::SetFlags(uint32 flags)	
<b>FSPDF_Annot_SetGroup</b>		foxit::pdf::Page::SetAnnotGroup(const annots::MarkupArray& annot_array, int header_index)	

5.3 C API		6.0 C++ API	
<b>FSPDF_Annot_SetHighlightingMode</b>		foxit::pdf::annots::Link::SetHighlightingMode(HighlightingMode mode) foxit::pdf::annots::Widget::SetHighlightingMode(HighlightingMode mode)	
<b>FSPDF_Annot_SetIconFit</b>		foxit::pdf::annots::Widget::SetMKIconFit(const IconFit& icon_fit)	
<b>FSPDF_Annot_SetIconName</b>		foxit::pdf::annots::Note::SetIconName(const char* icon_name) foxit::pdf::annots::Stamp::SetIconName(const char* icon_name) foxit::pdf::annots::FileAttachment::SetIconName(const char* icon_name)	
<b>FSPDF_Annot_SetIconProvider</b>		foxit::common::Library::SetAnnotIconProviderCallback(pdf::annots::IconProviderCallback* annot_icon_provider)	
<b>FSPDF_Annot_SetInkList</b>		foxit::pdf::annots::Ink::SetInkList(const common::Path& ink_list)	
<b>FSPDF_Annot_SetInnerRectMargin</b>		foxit::pdf::annots::Caret::SetInnerRect(const RectF& inner_rect) foxit::pdf::annots::Square::SetInnerRect(const RectF& inner_rect) foxit::pdf::annots::Circle::SetInnerRect(const RectF& inner_rect) foxit::pdf::annots::FreeText::SetInnerRect(const RectF& inner_rect)	
<b>FSPDF_Annot_SetIntent</b>		foxit::pdf::annots::Markup::SetIntent(const String& intent)	
<b>FSPDF_Annot_SetLeaderLineExtension</b>		foxit::pdf::annots::Line::SetLeaderLineExtensionLength(float extension_length)	
<b>FSPDF_Annot_SetLeaderLineLength</b>		foxit::pdf::annots::Line::SetLeaderLineLength(float length)	
<b>FSPDF_Annot_SetLeaderLineOffset</b>		foxit::pdf::annots::Line::SetLeaderLineOffset(float offset)	
<b>FSPDF_Annot_SetLineEndingStyles</b>		foxit::pdf::annots::Line::SetLineStartStyle(EndingStyle ending_style) foxit::pdf::annots::Line::SetLineEndStyle(EndingStyle ending_style) foxit::pdf::annots::PolyLine::SetLineStartStyle(EndingStyle starting_style) foxit::pdf::annots::PolyLine::SetLineEndStyle(EndingStyle ending_style) foxit::pdf::annots::FreeText::SetCalloutLineEndingStyle(EndingStyle ending_style)	
<b>FSPDF_Annot_SetLinePosition</b>		foxit::pdf::annots::Line::SetStartPoint(const PointF& point) foxit::pdf::annots::Line::SetEndPoint(const PointF& point)	
<b>FSPDF_Annot_SetMKCaption</b>		foxit::pdf::annots::Widget::SetMKNormalCaption(const wchar_t* caption) foxit::pdf::annots::Widget::SetMKRolloverCaption(const wchar_t* caption) foxit::pdf::annots::Widget::SetMKDownCaption(const wchar_t* caption)	
<b>FSPDF_Annot_SetMKColor</b>		foxit::pdf::annots::Widget::SetMKBorderColor(RGB color) foxit::pdf::annots::Widget::SetMKBackgroundColor(RGB color)	
<b>FSPDF_Annot_SetMKIconBitmap</b>		foxit::pdf::annots::Widget::SetMKNormalIconBitmap(const common::Bitmap& bitmap) foxit::pdf::annots::Widget::SetMKRolloverIconBitmap(const common::Bitmap& bitmap) foxit::pdf::annots::Widget::SetMKDownIconBitmap(const common::Bitmap& bitmap)	
<b>FSPDF_Annot_SetMKIconImage</b>		foxit::pdf::annots::Widget::SetMKNormalIconImage(const common::Image& image, int frame_index) foxit::pdf::annots::Widget::SetMKRolloverIconImage(const common::Image& image, int frame_index) foxit::pdf::annots::Widget::SetMKDownIconImage(const common::Image& image, int frame_index)	
<b>FSPDF_Annot_SetMKRotation</b>		foxit::pdf::annots::Widget::SetMKRotation(common::Rotation rotation)	
<b>FSPDF_Annot_SetModifiedDateTime</b>		foxit::pdf::annots::Annot::SetModifiedDateTime(const DateTime& date_time)	

5.3 C API		6.0 C++ API	
<b>FSPDF_Annot_SetName</b>		foxit::pdf::annots::Annot::SetUniqueID(const WString& unique_id)	
<b>FSPDF_Annot_SetOpacity</b>		foxit::pdf::annots::Markup::SetOpacity(float opacity) foxit::pdf::annots::Screen::SetOpacity(float opacity)	
<b>FSPDF_Annot_SetOpenStatus</b>		foxit::pdf::annots::Note::SetOpenStatus(bool status) foxit::pdf::annots::Popup::SetOpenStatus(bool status)	
<b>FSPDF_Annot_SetPopup</b>		foxit::pdf::annots::Markup::SetPopup(const Popup& popup)	
<b>FSPDF_Annot_SetQuadPoints</b>		foxit::pdf::annots::TextMarkup::SetQuadPoints(const QuadPointsArray& quad_points_array) foxit::pdf::annots::Link::SetQuadPoints(const QuadPointsArray& quad_points_array)	
<b>FSPDF_Annot_SetStamp</b>		foxit::pdf::annots::Stamp::SetImage(const common::Image& image, int frame_index, int compress)	
<b>FSPDF_Annot_SetStampWithBitmap</b>		foxit::pdf::annots::Stamp::SetBitmap(const common::Bitmap& bitmap)	
<b>FSPDF_Annot_SetState</b>		foxit::pdf::annots::Note::SetState(State state)	
<b>FSPDF_Annot_SetSubject</b>		foxit::pdf::annots::Markup::SetSubject(const WString& subject)	
<b>FSPDF_Annot_SetSymbol</b>		N/A	
<b>FSPDF_Annot_SetTitle</b>		foxit::pdf::annots::Markup::SetTitle(const WString& title)	
<b>FSPDF_Annot_SetVertices</b>		foxit::pdf::annots::Polygon::SetVertexes(const PointFArray& vertexes) foxit::pdf::annots::PolyLine::SetVertexes(const PointFArray& vertexes)	
<b>FSPDF_Annot_Ungroup</b>		foxit::pdf::annots::Markup::Ungroup()	
<b>FSPDF_Archive_DeserializeAnnot</b>		N/A	
<b>FSPDF_Archive_SerializeAnnot</b>		N/A	
<b>FSPDF_Array_Add</b>		foxit::pdf::objects::PDFArray::AddElement(PDFObject* element)	
<b>FSPDF_Array_AddBoolean</b>		foxit::pdf::objects::PDFArray::AddBoolean(bool boolean_value)	
<b>FSPDF_Array_AddDateTime</b>		foxit::pdf::objects::PDFArray::AddDateTime(const DateTime& date_time)	
<b>FSPDF_Array_AddFloat</b>		foxit::pdf::objects::PDFArray::AddFloat(float float_value)	
<b>FSPDF_Array_AddInteger</b>		foxit::pdf::objects::PDFArray::AddInteger(int integer_value)	
<b>FSPDF_Array_AddMatrix</b>		foxit::pdf::objects::PDFArray::AddMatrix(const Matrix& matrix)	
<b>FSPDF_Array_AddRawByteString</b>		N/A	
<b>FSPDF_Array_AddRect</b>		foxit::pdf::objects::PDFArray::AddRect(const RectF& rect)	
<b>FSPDF_Array_AddUnicodeName</b>		foxit::pdf::objects::PDFArray::AddName(const char* name)	
<b>FSPDF_Array_AddUnicodeString</b>		foxit::pdf::objects::PDFArray::AddString(const wchar_t* new_string)	
<b>FSPDF_Array_CountElements</b>		foxit::pdf::objects::PDFArray::GetElementCount() const	
<b>FSPDF_Array_GetArray</b>		foxit::pdf::objects::PDFArray::GetElement(int index) const foxit::pdf::objects::PDFObject::GetArray()	
<b>FSPDF_Array_GetBoolean</b>		foxit::pdf::objects::PDFArray::GetElement(int index) const foxit::pdf::objects::PDFObject::GetBoolean() const	
<b>FSPDF_Array_GetDateTime</b>		foxit::pdf::objects::PDFArray::GetElement(int index) const foxit::pdf::objects::PDFObject::GetDateTime() const	
<b>FSPDF_Array_GetDict</b>		foxit::pdf::objects::PDFArray::GetElement(int index) const foxit::pdf::objects::PDFObject::GetDict()	
<b>FSPDF_Array_GetElement</b>		foxit::pdf::objects::PDFArray::GetElement(int index) const	
<b>FSPDF_Array_GetFloat</b>		foxit::pdf::objects::PDFArray::GetElement(int index) const foxit::pdf::objects::PDFObject::GetFloat() const	

5.3 C API		6.0 C++ API	
<b>FSPDF_Array_GetInteger</b>		foxit::pdf::objects::PDFArray::GetElement(int index) const foxit::pdf::objects::PDFObject::GetInteger() const	
<b>FSPDF_Array_GetMatrix</b>		foxit::pdf::objects::PDFArray::GetElement(int index) const foxit::pdf::objects::PDFObject::GetMatrix() const	
<b>FSPDF_Array_GetRawByteString</b>		N/A	
<b>FSPDF_Array_GetRect</b>		foxit::pdf::objects::PDFArray::GetElement(int index) const foxit::pdf::objects::PDFObject::GetRect() const	
<b>FSPDF_Array_GetStream</b>		foxit::pdf::objects::PDFArray::GetElement(int index) const foxit::pdf::objects::PDFObject::GetStream()	
<b>FSPDF_Array_GetUnicodeString</b>		foxit::pdf::objects::PDFArray::GetElement(int index) const foxit::pdf::objects::PDFObject::GetWideString() const	
<b>FSPDF_Array_InsertAt</b>		foxit::pdf::objects::PDFArray::InsertAt(int index, PDFObject* element)	
<b>FSPDF_Array_RemoveAt</b>		foxit::pdf::objects::PDFArray::RemoveAt(int index)	
<b>FSPDF_Array_SetAt</b>		foxit::pdf::objects::PDFArray::SetAt(int index, PDFObject* element)	
<b>FSPDF_Attachment_Create</b>		foxit::pdf::FileSpec::FileSpec(const Doc& document)	
<b>FSPDF_Attachment_GetChecksum</b>		foxit::pdf::FileSpec::GetChecksum() const	
<b>FSPDF_Attachment_GetCreationDate ateTime</b>		foxit::pdf::FileSpec::GetCreationDateTime() const	
<b>FSPDF_Attachment_GetData</b>		foxit::pdf::FileSpec::GetFileData() const	
<b>FSPDF_Attachment_GetDescription n</b>		foxit::pdf::FileSpec::GetDescription()	
<b>FSPDF_Attachment_GetDocument</b>		N/A	
<b>FSPDF_Attachment_GetFileName</b>		foxit::pdf::FileSpec::GetFileName() const	
<b>FSPDF_Attachment_GetModifiedDate ateTime</b>		foxit::pdf::FileSpec::GetModifiedDateDateTime() const	
<b>FSPDF_Attachment_GetSize</b>		foxit::pdf::FileSpec::GetFileSize() const	
<b>FSPDF_Attachment_IsEmbedded</b>		foxit::pdf::FileSpec::IsEmbedded() const	
<b>FSPDF_Attachment_Release</b>		foxit::pdf::FileSpec::~FileSpec()	
<b>FSPDF_Attachment_SetChecksum</b>		foxit::pdf::FileSpec::SetChecksum(const String& checksum)	
<b>FSPDF_Attachment_SetCreationDate ateTime</b>		foxit::pdf::FileSpec::SetCreationDateTime(const DateTime& date_time)	
<b>FSPDF_Attachment_SetDescription n</b>		foxit::pdf::FileSpec::SetDescription(const String& description)	
<b>FSPDF_Attachment_SetFile</b>		foxit::pdf::FileSpec::Embed(const wchar_t* file_path)	
<b>FSPDF_Attachment_SetFileName</b>		foxit::pdf::FileSpec::SetFileName(const wchar_t* file_name)	
<b>FSPDF_Attachment_SetModifiedDate ateTime</b>		foxit::pdf::FileSpec::SetModifiedDateDateTime(const DateTime& date_time)	
<b>FSPDF_Attachment_WriteToFile</b>		foxit::pdf::FileSpec::ExportToFile(const wchar_t* path)	
<b>FSPDF_Attachments_CountAttach ment</b>		foxit::pdf::Attachments::GetCount()	
<b>FSPDF_Attachments_GetAttachm ent</b>		foxit::pdf::Attachments::GetKey(int index) foxit::pdf::Attachments::GetEmbeddedFile(const WString& key)	
<b>FSPDF_Attachments_InsertAttach ment</b>		foxit::pdf::Attachments::AddEmbeddedFile(const WString& key, const FileSpec& file_spec)	
<b>FSPDF_Attachments_Release</b>		foxit::pdf::Attachments::~Attachments()	

5.3 C API		6.0 C++ API	
<b>FSPDF_Attachments_RemoveAttachment</b>		foxit::pdf::Attachments::RemoveEmbeddedFile(const WString& key)	
<b>FSPDF_Bookmark_CloneIterator</b>		Not Required.	
<b>FSPDF_Bookmark_CountActions</b>		// Get directly related action foxit::pdf::Bookmark::GetAction() // Count the sub-action of a specific action. foxit::pdf::actions::Action::GetSubActionCount()	Every action may have sub-action, users can count them by yourself.
<b>FSPDF_Bookmark_GetAction</b>		// Get directly related action foxit::pdf::Bookmark::GetAction() // Get sub-action from action if sub-action exists foxit::pdf::actions::Action::GetSubAction(int index)	
<b>FSPDF_Bookmark_GetData</b>		foxit::pdf::Bookmark::GetTitle() const foxit::pdf::Bookmark::GetColor() const foxit::pdf::Bookmark::GetStyle() const	
<b>FSPDF_Bookmark_GetPos</b>		Not Required.	
<b>FSPDF_Bookmark_HasChild</b>		foxit::pdf::Bookmark::HasChild()	
<b>FSPDF_Bookmark_Insert</b>		foxit::pdf::Bookmark::Insert(const WString& title, Position position)	
<b>FSPDF_Bookmark_InsertAction</b>		// Get directly related action foxit::pdf::Bookmark::GetAction() // Insert a sub-action foxit::pdf::actions::Action::InsertSubAction(int index, const Action& sub_action)	
<b>FSPDF_Bookmark_InsertAtPos</b>		Not Required.	
<b>FSPDF_Bookmark_IsFirstChild</b>		foxit::pdf::Bookmark::IsFirstChild()	
<b>FSPDF_Bookmark_IsLastChild</b>		foxit::pdf::Bookmark::IsLastChild()	
<b>FSPDF_Bookmark_IsRoot</b>		foxit::pdf::Bookmark::IsRoot()	
<b>FSPDF_Bookmark_MoveToFirstChildId</b>		Not Required.	
<b>FSPDF_Bookmark_MoveToNextSibling</b>		Not Required.	
<b>FSPDF_Bookmark_MoveToParent</b>		Not Required.	
<b>FSPDF_Bookmark_MoveToPos</b>		Not Required.	
<b>FSPDF_Bookmark_MoveToPrevSibling</b>		Not Required.	
<b>FSPDF_Bookmark_MoveToRoot</b>		Not Required.	
<b>FSPDF_Bookmark_ReleaseIterator</b>		Not Required.	
<b>FSPDF_Bookmark_ReleasePos</b>		Not Required.	
<b>FSPDF_Bookmark_Remove</b>		foxit::pdf::Doc::RemoveBookmark(const Bookmark& bookmark)	

5.3 C API		6.0 C++ API	
<b>FSPDF_Bookmark_RemoveAction</b>		// Get directly related action foxit::pdf::Bookmark::GetAction() // Remove a specified sub-action if sub-action exists foxit::pdf::actions::Action::RemoveSubAction(int index)  // or remove directly related action foxit::pdf::Bookmark::RemoveAction()	
<b>FSPDF_Bookmark_RemoveAllActions</b>		foxit::pdf::Bookmark::RemoveAction()	
<b>FSPDF_Bookmark_SetAction</b>		// Get directly related action foxit::pdf::Bookmark::GetAction() // Set a specified sub-action if sub-action exists foxit::pdf::actions::Action::SetSubAction(int index, const Action& sub_action)  // or set directly related action foxit::pdf::Bookmark::SetAction(const actions::Action& action)	
<b>FSPDF_Bookmark_SetData</b>		foxit::pdf::Bookmark::SetTitle(const WString& title) foxit::pdf::Bookmark::SetColor(RGB color) foxit::pdf::Bookmark::SetStyle(uint32 style)	
<b>FSPDF_Bookmark_Update</b>		Not Required.	
<b>FSPDF_BookmarkData_Clear</b>		Not Required.	
<b>FSPDF_BookmarkData_Init</b>		Not Required.	
<b>FSPDF_ChoiceOption_Clear</b>		class foxit::pdf::interform::ChoiceOption	
<b>FSPDF_ChoiceOption_Init</b>		class foxit::pdf::interform::ChoiceOption	
<b>FSPDF_Dictionary_EnumEntry</b>		N/A	
<b>FSPDF_Dictionary_GetArray</b>		foxit::pdf::objects::PDFDictionary::GetElement(const char* key) const foxit::pdf::objects::PDFObject::GetArray()	
<b>FSPDF_Dictionary_GetBoolean</b>		foxit::pdf::objects::PDFDictionary::GetElement(const char* key) const foxit::pdf::objects::PDFObject::GetBoolean() const	
<b>FSPDF_Dictionary_GetDateTime</b>		foxit::pdf::objects::PDFDictionary::GetElement(const char* key) const foxit::pdf::objects::PDFObject::GetDateTime() const	
<b>FSPDF_Dictionary_GetDict</b>		foxit::pdf::objects::PDFDictionary::GetElement(const char* key) const foxit::pdf::objects::PDFObject::GetDict()	
<b>FSPDF_Dictionary_GetElement</b>		foxit::pdf::objects::PDFDictionary::GetElement(const char* key) const	
<b>FSPDF_Dictionary_GetFloat</b>		foxit::pdf::objects::PDFDictionary::GetElement(const char* key) const foxit::pdf::objects::PDFObject::GetFloat() const	
<b>FSPDF_Dictionary_GetInteger</b>		foxit::pdf::objects::PDFDictionary::GetElement(const char* key) const foxit::pdf::objects::PDFObject::GetInteger() const	
<b>FSPDF_Dictionary_GetMatrix</b>		foxit::pdf::objects::PDFDictionary::GetElement(const char* key) const foxit::pdf::objects::PDFObject::GetMatrix() const	
<b>FSPDF_Dictionary_GetRawByteString</b>		N/A	
<b>FSPDF_Dictionary_GetRect</b>		foxit::pdf::objects::PDFDictionary::GetElement(const char* key) const foxit::pdf::objects::PDFObject::GetRect() const	
<b>FSPDF_Dictionary_GetStream</b>		foxit::pdf::objects::PDFDictionary::GetElement(const char* key) const foxit::pdf::objects::PDFObject::GetStream()	

5.3 C API		6.0 C++ API	
<b>FSPDF_Dictionary_GetUnicodeString</b>		foxit::pdf::objects::PDFDictionary::GetElement(const char* key) const foxit::pdf::objects::PDFObject::GetWideString() const	
<b>FSPDF_Dictionary_HasKey</b>		foxit::pdf::objects::PDFDictionary::HasKey(const char* key) const	
<b>FSPDF_Dictionary_RemoveAt</b>		foxit::pdf::objects::PDFDictionary::RemoveAt(const char* key)	
<b>FSPDF_Dictionary_SetAt</b>		foxit::pdf::objects::PDFDictionary::SetAt(const char* key, PDFObject* object)	
<b>FSPDF_Dictionary_SetAtBoolean</b>		foxit::pdf::objects::PDFDictionary::SetAtBoolean(const char* key, bool value)	
<b>FSPDF_Dictionary_SetAtDateTime</b>		foxit::pdf::objects::PDFDictionary::SetAtDateTime(const char* key, const DateTime& value)	
<b>FSPDF_Dictionary_SetAtFloat</b>		foxit::pdf::objects::PDFDictionary::SetAtFloat(const char* key, float value)	
<b>FSPDF_Dictionary_SetAtInteger</b>		foxit::pdf::objects::PDFDictionary::SetAtInteger(const char* key, int value)	
<b>FSPDF_Dictionary_SetAtMatrix</b>		foxit::pdf::objects::PDFDictionary::SetAtMatrix(const char* key, const Matrix& value)	
<b>FSPDF_Dictionary_SetAtRawByteString</b>		N/A	
<b>FSPDF_Dictionary_SetAtRect</b>		foxit::pdf::objects::PDFDictionary::SetAtRect(const char* key, const RectF& value)	
<b>FSPDF_Dictionary_SetAtUnicodeName</b>		foxit::pdf::objects::PDFDictionary::SetName(const char* key, const char* value)	
<b>FSPDF_Dictionary_SetAtUnicodeString</b>		foxit::pdf::objects::PDFDictionary::SetString(const char* key, const wchar_t* value)	
<b>FSPDF_Doc_AddLayer</b>		N/A	
<b>FSPDF_Doc_AsyncLoad</b>		foxit::pdf::Doc::Doc(common::file::ReaderCallback* file_read, bool is_async = false) foxit::common::file::AsyncReaderCallback foxit::pdf::Doc::Load or StartLoad	
<b>FSPDF_Doc_ClearCache</b>		foxit::pdf::Doc::ClearRenderCache()	
<b>FSPDF_Doc_Close</b>		foxit::pdf::Doc::~Doc()	
<b>FSPDF_Doc_CountActions</b>		// Get directly related action foxit::pdf::Doc::GetOpenAction() or foxit::pdf::actions::AdditionalAction::AdditionalAction(const Doc& doc, objects::PDFDictionary* pdf_dict = NULL) foxit::pdf::actions::AdditionalAction::GetAction(TriggerEvent trigger) // Count the sub-action of a specific action. foxit::pdf::actions::Action::GetSubActionCount(int index)	Every action may have sub-action, users can count them by yourself.
<b>FSPDF_Doc_CountFonts</b>		foxit::pdf::Doc::GetFontCount()	
<b>FSPDF_Doc_CountPages</b>		foxit::pdf::Doc::GetPageCount() const	
<b>FSPDF_Doc_CountSignatures</b>		foxit::pdf::Doc::GetSignatureCount()	
<b>FSPDF_Doc_Create</b>		foxit::pdf::Doc::Doc()	
<b>FSPDF_Doc_CreateBookmarkIterator</b>		Not Required.	
<b>FSPDF_Doc_CreateForm</b>		foxit::pdf::interform::Form::Form(const pdf::Doc& document)	
<b>FSPDF_Doc_DoJavaScriptAction</b>		foxit::pdf::actions::AdditionalAction::AdditionalAction(const Doc& doc, objects::PDFDictionary* pdf_dict = NULL) foxit::pdf::actions::AdditionalAction::DoJSAction(TriggerEvent trigger)	

5.3 C API		6.0 C++ API	
<b>FSPDF_Doc_EmbedFont</b>		N/A	
<b>FSPDF_Doc_EnumAllPageSize</b>	deprecated	Not Required.	
<b>FSPDF_Doc_EnumLayers</b>		foxit::pdf::LayerTree::LayerTree(const Doc& document)	
<b>FSPDF_Doc_EnumPagesInfo</b>		foxit::pdf::Doc::GetPageBasicInfo(int index)	Get the size and rotation of all the pages through recursively calling this function.
<b>FSPDF_Doc_ExportAnnotsToFDFDoc</b>		foxit::pdf::Doc::ExportToFDF(const fdf::Doc& fdf_doc, int types = e_Forms   e_Annots, const common::Range& page_range = common::Range())	
<b>FSPDF_Doc_GetAction</b>		// Get directly related action foxit::pdf::Doc::GetOpenAction() or foxit::pdf::actions::AdditionalAction::AdditionalAction(const Doc& doc, objects::PDFDictionary* pdf_dict = NULL) foxit::pdf::actions::AdditionalAction::GetAction(TriggerEvent trigger) // Get a specified sub-action if sub-action exists foxit::pdf::actions::Action::GetSubAction(int index)	
<b>FSPDF_Doc_GetCatalog</b>		foxit::pdf::Doc::GetCatalog() const	
<b>FSPDF_Doc_GetDisplayMode</b>		foxit::pdf::Doc::GetDisplayMode() const	
<b>FSPDF_Doc_GetEncryptDict</b>		foxit::pdf::Doc::GetEncryptDict() const	
<b>FSPDF_Doc_GetEncryptionFilter</b>		foxit::pdf::Doc::GetStdEncryptData() const foxit::pdf::Doc::GetCertificateEncryptData() const foxit::pdf::Doc::GetDRMEncryptData() const foxit::pdf::Doc::GetCustomEncryptData() const foxit::pdf::Doc::GetRMSEncryptData() const	
<b>FSPDF_Doc_GetEncryptionParam</b>		foxit::pdf::Doc::GetStdEncryptData() const foxit::pdf::Doc::GetCertificateEncryptData() const foxit::pdf::Doc::GetDRMEncryptData() const foxit::pdf::Doc::GetCustomEncryptData() const foxit::pdf::Doc::GetRMSEncryptData() const	
<b>FSPDF_Doc_GetEncryptionSubFilter</b>		foxit::pdf::Doc::GetStdEncryptData() const foxit::pdf::Doc::GetCertificateEncryptData() const foxit::pdf::Doc::GetDRMEncryptData() const foxit::pdf::Doc::GetCustomEncryptData() const foxit::pdf::Doc::GetRMSEncryptData() const	
<b>FSPDF_Doc_GetEncryptionType</b>		foxit::pdf::Doc::GetEncryptionType() const	
<b>FSPDF_Doc_GetFileVersion</b>		foxit::pdf::Doc::GetFileVersion()	
<b>FSPDF_Doc_GetFirstAvailPageIndex</b>		foxit::pdf::Doc::GetFirstAvailPageIndex() const	
<b>FSPDF_Doc_GetFontByIndex</b>		foxit::pdf::Doc::GetFont(int index)	
<b>FSPDF_Doc_GetInfoDict</b>		foxit::pdf::Doc::GetInfo() const	
<b>FSPDF_Doc_GetJSActionByName</b>		foxit::pdf::objects::PDFNameTree::PDFNameTree(const Doc& document, Type type) with type=e_JavaScript foxit::pdf::objects::PDFNameTree::GetObj(const WString& name) foxit::pdf::actions::Action::Action(const Doc& document, objects::PDFDictionary* action_dict)	



5.3 C API		6.0 C++ API	
<b>FSPDF_Doc_GetPage</b>		foxit::pdf::Doc::GetPage(int index)	
<b>FSPDF_Doc_GetPasswordType</b>		foxit::pdf::Doc::GetPasswordType() const	
<b>FSPDF_Doc_GetSignature</b>		foxit::pdf::Doc::GetSignature(int index)	
<b>FSPDF_Doc_GetTrailer</b>		foxit::pdf::Doc::GetTrailer() const	
<b>FSPDF_Doc_GetUserPermissions</b>		foxit::pdf::Doc::GetUserPermissions() const	
<b>FSPDF_Doc_GetWrapperData</b>		foxit::pdf::Doc::GetWrapperData() const	
<b>FSPDF_Doc_GetWrapperOffset</b>		foxit::pdf::Doc::GetWrapperOffset() const	
<b>FSPDF_Doc_HasForm</b>		foxit::pdf::Doc::HasForm() const	
<b>FSPDF_Doc_InitiateJavaScript</b>		foxit::pdf::Doc::DoJSOpenAction()	
<b>FSPDF_Doc_InsertAction</b>		// Get directly related action foxit::pdf::Doc::GetOpenAction() or foxit::pdf::actions::AdditionalAction::AdditionalAction(const Doc& doc, objects::PDFDictionary* pdf_dict = NULL) foxit::pdf::actions::AdditionalAction::GetAction(TriggerEvent trigger) // Insert a sub-action foxit::pdf::actions::Action::InsertSubAction(int index, const Action& sub_action)	
<b>FSPDF_Doc_IsDocAvail</b>		Not Required.	
<b>FSPDF_Doc_IsFormAvail</b>		Not Required.	
<b>FSPDF_Doc_IsLinearized</b>		foxit::pdf::Doc::IsLinearized() const	
<b>FSPDF_Doc_IsModified</b>		N/A	
<b>FSPDF_Doc_IsPageAvail</b>		Not Required.	
<b>FSPDF_Doc_IsPortfolio</b>		foxit::pdf::Doc::IsPortfolio()	
<b>FSPDF_Doc_IsWrapper</b>		foxit::pdf::Doc::IsWrapper() const	
<b>FSPDF_Doc_LoadAttachments</b>		foxit::pdf::Attachments::Attachments(const Doc& doc, const objects::PDFNameTree& nametree = objects::PDFNameTree())	
<b>FSPDF_Doc_PageIndexToPageLabel</b>		foxit::pdf::PageLabels::PageLabels(const Doc& document) foxit::pdf::PageLabels::GetPageLabelTitle(int page_index) const	
<b>FSPDF_Doc_PageLabelToPageIndex</b>		N/A	
<b>FSPDF_Doc_RemoveAction</b>		// Get directly related action foxit::pdf::Doc::GetOpenAction() or foxit::pdf::actions::AdditionalAction::AdditionalAction(const Doc& doc, objects::PDFDictionary* pdf_dict = NULL) foxit::pdf::actions::AdditionalAction::GetAction(TriggerEvent trigger) // Remove a specified sub-action if sub-action exists foxit::pdf::actions::Action::RemoveSubAction(int index)  // or remove directly related action foxit::pdf::Doc::RemoveOpenAction() foxit::pdf::actions::AdditionalAction::RemoveAction(TriggerEvent trigger)	
<b>FSPDF_Doc_RemoveAllActions</b>		foxit::pdf::Doc::RemoveOpenAction() foxit::pdf::actions::AdditionalAction::AdditionalAction(const Doc& doc, objects::PDFDictionary* pdf_dict = NULL) foxit::pdf::actions::AdditionalAction::RemoveAllActions()	

5.3 C API		6.0 C++ API	
<b>FSPDF_Doc_RemoveJSActionByName</b>		foxit::pdf::objects::PDFNameTree::PDFNameTree(const Doc& document, Type type) with type=e_JavaScript foxit::pdf::objects::PDFNameTree::RemoveObj(const WString& name)	
<b>FSPDF_Doc_SaveAsWrapperFile</b>		foxit::pdf::Doc::SaveAsWrapperFile(const wchar_t* file_path, const WrapperData* wrapper_data = NULL, uint32 user_permissions = 0xFFFFFFFF, const char* owner_password = "")	
<b>FSPDF_Doc_SetAction</b>		// Get directly related action foxit::pdf::Doc::GetOpenAction() or foxit::pdf::actions::AdditionalAction::AdditionalAction(const Doc& doc, objects::PDFDictionary* pdf_dict = NULL) foxit::pdf::actions::AdditionalAction::GetAction(TriggerEvent trigger) // Set a specified sub-action if sub-action exists foxit::pdf::actions::Action::SetSubAction(int index, const Action& sub_action)  // or set directly related action foxit::pdf::Doc::SetOpenAction(actions::Action& action) foxit::pdf::actions::AdditionalAction::SetAction(TriggerEvent trigger, const Action& action)	
<b>FSPDF_Doc_SetActionHandler</b>	deprecated	Not Required.	
<b>FSPDF_Doc_SetDisplayMode</b>		foxit::pdf::Doc::SetDisplayMode(DisplayMode display_mode)	
<b>FSPDF_Doc_SetFileVersion</b>		foxit::pdf::Doc::SetFileVersion(int version)	
<b>FSPDF_Doc_SetJavaScriptActionHandler</b>		foxit::common::Library::SetActionCallback(ActionCallback* action_callback)	
<b>FSPDF_Doc_SetLayersBaseState</b>		foxit::pdf::LayerTree::LayerTree(const Doc& document) foxit::pdf::LayerTree::SetBaseState(UsageState state)	
<b>FSPDF_Doc_StartImportPages</b>		foxit::pdf::Doc::StartImportPages(int dest_index, const Doc& src_doc, uint32 flags = Doc::e_ImportFlagNormal, const char* layer_name = "", const common::Range& page_range = common::Range(), common::PauseCallback* pause = NULL)	
<b>FSPDF_Doc_StartImportPagesFromFileName</b>		foxit::pdf::Doc::StartImportPagesFromFilePath(int dest_index, const wchar_t* src_file_path, const WString& password, uint32 flags = Doc::e_ImportFlagNormal, const char* layer_name = "", const common::Range& page_range = common::Range(), common::PauseCallback* pause = NULL)	
<b>FSPDF_Doc_StartImportPagesWithLayers</b>		foxit::pdf::Doc::StartImportPages(int dest_index, const Doc& src_doc, uint32 flags = Doc::e_ImportFlagNormal, const char* layer_name = "", const common::Range& page_range = common::Range(), common::PauseCallback* pause = NULL)	
<b>FSPDF_Doc_StartLoad</b>		foxit::pdf::Doc::StartLoad(const String& password = "", bool is_cache_stream = true, common::PauseCallback* pause = NULL) foxit::pdf::Doc::StartLoadW(const WString& password = WString(), bool is_cache_stream = true, common::PauseCallback* pause = NULL)	
<b>FSPDF_Doc_StartLoadWithFlag</b>		foxit::pdf::Doc::StartLoad(const String& password = "", bool is_cache_stream = true, common::PauseCallback* pause = NULL) foxit::pdf::Doc::StartLoadW(const WString& password = WString(), bool is_cache_stream = true, common::PauseCallback* pause = NULL)	
<b>FSPDF_Doc_StartReload</b>		N/A	

5.3 C API		6.0 C++ API	
<b>FSPDF_Doc_StartSaveToFile</b>		foxit::pdf::Doc::StartSaveAs(const char* file_path, uint32 save_flags = Doc::e_SaveFlagNormal, common::PauseCallback* pause = NULL) foxit::pdf::Doc::StartSaveAs(const wchar_t* file_path, uint32 save_flags = Doc::e_SaveFlagNormal, common::PauseCallback* pause = NULL) foxit::pdf::Doc::StartSaveAs(common::file::WriterCallback* file, uint32 save_flags = Doc::e_SaveFlagNormal, common::PauseCallback* pause = NULL)	
<b>FSPDF_Font_GetDict</b>		N/A	
<b>FSPDF_Font_IsEmbedded</b>		foxit::common::Font::IsEmbedded(const pdf::Doc& document)	
<b>FSPDF_Font_IsSupportEmbedded</b>		N/A	
<b>FSPDF_Form_AddField</b>		foxit::pdf::interform::Form::AddControl(const Page& page, const wchar_t* field_name, Field::Type field_type, const RectF& rect)	
<b>FSPDF_Form_CountFields</b>		foxit::pdf::interform::Form::GetFieldCount(const wchar_t* filter = Wstring())	
<b>FSPDF_Form_ExportToFDFDoc</b>		foxit::pdf::Doc::ExportToFDF(const fdf::Doc& fdf_doc, int types = e_Forms   e_Annots, const common::Range& page_range = common::Range())	
<b>FSPDF_Form_ExportToXML</b>		foxit::pdf::interform::Form::ExportToXML(const char* file_path)	
<b>FSPDF_Form_GetAlignment</b>		foxit::pdf::interform::Form::GetAlignment() const	
<b>FSPDF_Form_GetControlAtDevice Pos</b>		foxit::pdf::interform::Form::GetControlAtDevicePoint(const Page& page, const PointF& point, float tolerance, Field::Type type, const Matrix* matrix = NULL)	
<b>FSPDF_Form_GetControlAtPos</b>		foxit::pdf::interform::Form::GetControlAtPoint(const Page& page, const PointF& point, float tolerance, Field::Type type)	
<b>FSPDF_Form_GetDefaultAppearance</b>		foxit::pdf::interform::Form::GetDefaultAppearance() const	
<b>FSPDF_Form_GetDocument</b>		foxit::pdf::interform::Form::GetDocument() const	
<b>FSPDF_Form_GetField</b>		foxit::pdf::interform::Form::GetField(int index, const wchar_t* filter = Wstring())	
<b>FSPDF_Form_GetFieldsInCalculationOrder</b>		foxit::pdf::interform::Form::GetFieldsInCalculationOrder()	
<b>FSPDF_Form_ImportFromFDFDoc</b>		foxit::pdf::Doc::ImportFromFDF(const fdf::Doc& fdf_doc, int types = e_Forms   e_Annots, const common::Range& page_range = common::Range())	
<b>FSPDF_Form_ImportFromXML</b>		foxit::pdf::interform::Form::ImportFromXML(const char* file_path)	
<b>FSPDF_Form_Load</b>		foxit::pdf::interform::Form::Form(const pdf::Doc& document)	
<b>FSPDF_Form_NeedConstructAppearances</b>		foxit::pdf::interform::Form::NeedConstructAppearances() const	
<b>FSPDF_Form_Release</b>		foxit::pdf::interform::Form::~~Form()	
<b>FSPDF_Form_RemoveField</b>		foxit::pdf::interform::Form::RemoveField(Field& field)	
<b>FSPDF_Form_RenameField</b>		foxit::pdf::interform::Form::RenameField(Field& field, const wchar_t* new_field_name)	
<b>FSPDF_Form_Reset</b>		foxit::pdf::interform::Form::Reset()	
<b>FSPDF_Form_SetAlignment</b>		foxit::pdf::interform::Form::SetAlignment(common::Alignment alignment)	
<b>FSPDF_Form_SetConstructAppearancesFlag</b>		foxit::pdf::interform::Form::SetConstructAppearances(bool need_construct)	
<b>FSPDF_Form_SetDefaultAppearance</b>		foxit::pdf::interform::Form::SetDefaultAppearance(const DefaultAppearance& default_ap)	
<b>FSPDF_Form_SetFieldsInCalculationOrder</b>		foxit::pdf::interform::Form::SetFieldsInCalculationOrder(const FieldArray& field_array)	

5.3 C API		6.0 C++ API	
<b>FSPDF_Form_ValidateFieldName</b>		foxit::pdf::interform::Form::ValidateFieldName(Field::Type type, const WString& field_name)	
<b>FSPDF_FormControl_DoJavaScript Action</b>		foxit::pdf::actions::AdditionalAction::AdditionalAction(const interform::Field& field) foxit::pdf::actions::AdditionalAction::DoJSAction(TriggerEvent trigger)	
<b>FSPDF_FormControl_GetExportValue</b>		foxit::pdf::interform::Control::GetExportValue() const	
<b>FSPDF_FormControl_GetFieldName</b>		foxit::pdf::interform::Control::GetField() const foxit::pdf::interform::Field::GetName() const	
<b>FSPDF_FormControl_GetWidgetAnnotation</b>		foxit::pdf::interform::Control::GetWidget() const	
<b>FSPDF_FormControl_IsChecked</b>		foxit::pdf::interform::Control::IsChecked() const	
<b>FSPDF_FormControl_IsDefaultChecked</b>		foxit::pdf::interform::Control::IsDefaultChecked() const	
<b>FSPDF_FormControl_SetChecked</b>		foxit::pdf::interform::Control::SetChecked(bool checked)	
<b>FSPDF_FormControl_SetDefaultChecked</b>		foxit::pdf::interform::Control::SetDefaultChecked(bool checked)	
<b>FSPDF_FormControl_SetExportValue</b>		foxit::pdf::interform::Control::SetExportValue(const wchar_t* value)	
<b>FSPDF_FormField_Calculate</b>		N/A	
<b>FSPDF_FormField_CountActions</b>		foxit::pdf::actions::AdditionalAction::AdditionalAction(const interform::Field& field) foxit::pdf::actions::AdditionalAction::GetAction(TriggerEvent trigger) // Count the sub-action of a specific action foxit::pdf::actions::Action::GetSubActionCount()	Every action may have sub-action, users can count them by yourself.
<b>FSPDF_FormField_CountControls</b>		foxit::pdf::interform::Field::GetControlCount(const Page& page) const	
<b>FSPDF_FormField_Format</b>		N/A	
<b>FSPDF_FormField_FormatEvent</b>		N/A	
<b>FSPDF_FormField_GetAction</b>		foxit::pdf::actions::AdditionalAction::AdditionalAction(const interform::Field& field) foxit::pdf::actions::AdditionalAction::GetAction(TriggerEvent trigger) // Get a specified sub-action if sub-action exists foxit::pdf::actions::Action::GetSubAction(int index)	
<b>FSPDF_FormField_GetAlignment</b>		foxit::pdf::interform::Field::GetAlignment() const	
<b>FSPDF_FormField_GetAlternateName</b>		foxit::pdf::interform::Field::GetAlternateName() const	
<b>FSPDF_FormField_GetControl</b>		foxit::pdf::interform::Field::GetControl(const Page& page, int index)	
<b>FSPDF_FormField_GetControlIndex</b>		foxit::pdf::interform::Control::GetIndex() const	
<b>FSPDF_FormField_GetDefaultAppearance</b>		foxit::pdf::interform::Field::GetDefaultAppearance() const	
<b>FSPDF_FormField_GetDefaultValue</b>		foxit::pdf::interform::Field::GetDefaultValue() const	
<b>FSPDF_FormField_GetFlags</b>		foxit::pdf::interform::Field::GetFlags() const	
<b>FSPDF_FormField_GetMappingName</b>		foxit::pdf::interform::Field::GetMappingName() const	
<b>FSPDF_FormField_GetMaxLen</b>		foxit::pdf::interform::Field::GetMaxLength() const	
<b>FSPDF_FormField_GetOptions</b>		foxit::pdf::interform::Field::GetOptions() const	

5.3 C API		6.0 C++ API	
<b>FSPDF_FormField_GetTopVisibleIndex</b>		foxit::pdf::interform::Field::GetTopVisibleIndex()	
<b>FSPDF_FormField_GetType</b>		foxit::pdf::interform::Field::GetType() const	
<b>FSPDF_FormField_GetValue</b>		foxit::pdf::interform::Field::GetValue() const	
<b>FSPDF_FormField_InsertAction</b>		foxit::pdf::actions::AdditionalAction::AdditionalAction(const interform::Field& field) foxit::pdf::actions::AdditionalAction::GetAction(TriggerEvent trigger) // Insert a sub-action foxit::pdf::actions::Action::InsertSubAction(int index, const Action& sub_action)	
<b>FSPDF_FormField_RemoveAction</b>		foxit::pdf::actions::AdditionalAction::AdditionalAction(const interform::Field& field) foxit::pdf::actions::AdditionalAction::GetAction(TriggerEvent trigger) // Remove a specified sub-action if sub-action exists foxit::pdf::actions::Action::RemoveSubAction(int index)  // or remove directly related action foxit::pdf::actions::AdditionalAction::RemoveAction(TriggerEvent trigger)	
<b>FSPDF_FormField_RemoveAllActions</b>		foxit::pdf::actions::AdditionalAction::AdditionalAction(const interform::Field& field) foxit::pdf::actions::AdditionalAction::RemoveAllActions()	
<b>FSPDF_FormField_RemoveControl</b>		foxit::pdf::interform::Form::RemoveControl(Control& control)	
<b>FSPDF_FormField_SetAction</b>		foxit::pdf::actions::AdditionalAction::AdditionalAction(const interform::Field& field) foxit::pdf::actions::AdditionalAction::GetAction(TriggerEvent trigger) // Set a specified sub-action if sub-action exists foxit::pdf::actions::Action::SetSubAction(int index, const Action& sub_action)  // or set directly related action foxit::pdf::actions::AdditionalAction::SetAction(TriggerEvent trigger,const Action& action)	
<b>FSPDF_FormField_SetAlignment</b>		foxit::pdf::interform::Field::SetAlignment(common::Alignment alignment)	
<b>FSPDF_FormField_SetAlternateName</b>		foxit::pdf::interform::Field::SetAlternateName(const wchar_t* alternate_name)	
<b>FSPDF_FormField_SetDefaultAppearance</b>		foxit::pdf::interform::Field::SetDefaultAppearance(const DefaultAppearance& default_ap)	
<b>FSPDF_FormField_SetDefaultValue</b>		foxit::pdf::interform::Field::SetDefaultValue(const wchar_t* value)	
<b>FSPDF_FormField_SetFlags</b>		foxit::pdf::interform::Field::SetFlags(uint32 flags)	
<b>FSPDF_FormField_SetMappingName</b>		foxit::pdf::interform::Field::SetMappingName(const wchar_t* name)	
<b>FSPDF_FormField_SetMaxLen</b>		foxit::pdf::interform::Field::SetMaxLength(int max_length)	
<b>FSPDF_FormField_SetOptions</b>		foxit::pdf::interform::Field::SetOptions(const ChoiceOptionArray& option_array)	
<b>FSPDF_FormField_SetTopVisibleIndex</b>		foxit::pdf::interform::Field::SetTopVisibleIndex(int index)	
<b>FSPDF_FormField_SetValue</b>		foxit::pdf::interform::Field::SetValue(const wchar_t* value)	
<b>FSPDF_FormField_ValidateKeyStroke</b>		N/A	

5.3 C API		6.0 C++ API	
<b>FSPDF_FormField_ValidateKeyStrokeEvent</b>		N/A	
<b>FSPDF_FormField_ValidateValue</b>		N/A	
<b>FSPDF_FormField_ValidateValueEvent</b>		N/A	
<b>FSPDF_FormFiller_Begin</b>		foxit::pdf::interform::Filler::Filler(const Form& form, FillerAssistCallback* assist)	
<b>FSPDF_FormFiller_End</b>		foxit::pdf::interform::Filler::~Filler()	
<b>FSPDF_FormFiller_SetHighlightColor</b>		foxit::pdf::interform::Filler::SetHighlightColor(ARGB color)	
<b>FSPDF_FormFiller_ShowHighlight</b>		foxit::pdf::interform::Filler::HighlightFormFields(bool is_highlight)	
<b>FSPDF_FormFiller_TriggerWindowlessEvent</b>		foxit::pdf::interform::Filler::OnLButtonDown(const Page& page, const PointF& point, uint32 flags) foxit::pdf::interform::Filler::OnLButtonUp(const Page& page, const PointF& point, uint32 flags) foxit::pdf::interform::Filler::OnLButtonDoubleClick(const Page& page, const PointF& point, uint32 flags) foxit::pdf::interform::Filler::OnMouseMove(const Page& page, const PointF& point, uint32 flags) foxit::pdf::interform::Filler::OnMouseEnter(const Page& page, const PointF& point, uint32 flags) foxit::pdf::interform::Filler::OnMouseHover(const Page& page, const PointF& point, uint32 flags) foxit::pdf::interform::Filler::OnMouseLeave(const Page& page, const PointF& point, uint32 flags) foxit::pdf::interform::Filler::OnMouseWheel(const Page& page, const PointF& point, int32 delta, uint32 flags) foxit::pdf::interform::Filler::OnRButtonDown(const Page& page, const PointF& point, uint32 flags) foxit::pdf::interform::Filler::OnRButtonUp(const Page& page, const PointF& point, uint32 flags) foxit::pdf::interform::Filler::OnRButtonDoubleClick(const Page& page, const PointF& point, uint32 flags) foxit::pdf::interform::Filler::OnWheelButtonDown(const Page& page, const PointF& point, uint32 flags) foxit::pdf::interform::Filler::OnWheelButtonUp(const Page& page, const PointF& point, uint32 flags) foxit::pdf::interform::Filler::OnWheelButtonDoubleClick(const Page& page, const PointF& point, uint32 flags) foxit::pdf::interform::Filler::OnKeyDown(uint32 key_code, uint32 flags) foxit::pdf::interform::Filler::OnKeyUp(uint32 key_code, uint32 flags) foxit::pdf::interform::Filler::OnChar(uint32 char_code, uint32 flags) foxit::pdf::interform::Filler::SetFocus(const Control& control) foxit::pdf::interform::Filler::KillFocus()	
<b>FSPDF_FormXObject_Create</b>		foxit::pdf::graphics::FormXObject::Create(const Doc& document)	
<b>FSPDF_FormXObject_ExtractAPFromPage</b>		foxit::pdf::graphics::FormXObject::ImportPageContent(const Page& source_page, bool is_annots_included)	
<b>FSPDF_FormXObject_GetObjects</b>		foxit::pdf::graphics::FormXObject::GetGraphicsObjects() const	
<b>FSPDF_FormXObject_GetStream</b>		foxit::pdf::graphics::FormXObject::GetStream() const	
<b>FSPDF_ImageObject_CloneBitmap</b>		foxit::pdf::graphics::ImageObject::CloneBitmap(const GraphicsObjects& graphics_objects)	
<b>FSPDF_ImageObject_Create</b>		foxit::pdf::graphics::ImageObject::Create(const Doc& document)	

5.3 C API		6.0 C++ API	
<b>FSPDF_ImageObject_GetColorSpace</b>		foxit::pdf::graphics::ImageObject::GetColorSpace() const	
<b>FSPDF_ImageObject_GetStream</b>		foxit::pdf::graphics::ImageObject::GetStream() const	
<b>FSPDF_ImageObject_SetBitmap</b>		foxit::pdf::graphics::ImageObject::SetBitmap(const common::Bitmap& bitmap, const common::Bitmap& mask = common::Bitmap())	
<b>FSPDF_ImageObject_SetImage</b>		foxit::pdf::graphics::ImageObject::SetImage(const common::Image& image, int frame_index)	
<b>FSPDF_ImageObject_SetImageFromFile</b>		foxit::common::Image::Image(const char* path) or foxit::common::Image::Image(const wchar_t* path) Then Foxit::pdf::graphics::ImageObject::SetImage(const common::Image& image, int frame_index)	
<b>FSPDF_Layer_AddPageObject</b>		foxit::pdf::LayerNode::AddGraphicsObject(const Page& page, graphics::GraphicsObject* graphicsobject)	
<b>FSPDF_Layer_EnumeratePageObject</b>		foxit::pdf::LayerNode::GetGraphicsObjects(const Page& page)	
<b>FSPDF_Layer_GetApplicationUsage</b>		foxit::pdf::LayerNode::GetViewUsage() foxit::pdf::LayerNode::GetExportUsage() foxit::pdf::LayerNode::GetPrintUsage() foxit::pdf::LayerNode::GetZoomUsage()	
<b>FSPDF_Layer_GetName</b>		foxit::pdf::LayerNode::GetName()	
<b>FSPDF_Layer_HasIntent</b>		foxit::pdf::LayerNode::HasIntent(const char* intent)	
<b>FSPDF_Layer_IsInPage</b>		foxit::pdf::LayerNode::IsInPage(const Page& page)	
<b>FSPDF_Layer_Remove</b>		N/A	
<b>FSPDF_Layer_RemoveApplicationUsage</b>		foxit::pdf::LayerNode::RemoveUsage(LayerContext::UsageType usage_type)	
<b>FSPDF_Layer_RemovePageObject</b>		foxit::pdf::LayerNode::RemoveGraphicsObject(graphics::GraphicsObject* graphics_object)	
<b>FSPDF_Layer_SetApplicationUsage</b>		foxit::pdf::LayerNode::SetViewUsage(LayerTree::UsageState state) foxit::pdf::LayerNode::SetExportUsage(LayerTree::UsageState state) foxit::pdf::LayerNode::SetPrintUsage(const LayerPrintData& data) foxit::pdf::LayerNode::SetZoomUsage(const LayerZoomData& data)	
<b>FSPDF_Layer_SetDefaultState</b>		foxit::pdf::LayerNode::SetDefaultVisible(bool is_visible)	
<b>FSPDF_Layer_SetName</b>		foxit::pdf::LayerNode::SetName(const wchar_t* name)	
<b>FSPDF_LayerContext_CopyStates</b>		foxit::pdf::LayerContext::CopyStates(const LayerContext& source_context)	
<b>FSPDF_LayerContext_Create</b>		foxit::pdf::LayerContext::LayerContext(const Doc& document, UsageType usage_type)	
<b>FSPDF_LayerContext_GetDocument</b>		foxit::pdf::LayerContext::GetDocument()	
<b>FSPDF_LayerContext_GetUsageType</b>		foxit::pdf::LayerContext::GetUsageType()	
<b>FSPDF_LayerContext_IsVisible</b>		foxit::pdf::LayerContext::IsVisible(const LayerNode& layer)	
<b>FSPDF_LayerContext_MergeStates</b>		foxit::pdf::LayerContext::MergeStates(const LayerContext& source_context)	
<b>FSPDF_LayerContext_Release</b>		foxit::pdf::LayerContext::~~LayerContext()	
<b>FSPDF_LayerContext_Reset</b>		foxit::pdf::LayerContext::Reset()	
<b>FSPDF_LayerContext_SetVisible</b>		foxit::pdf::LayerContext::SetVisible(const LayerNode& layer, bool is_visible)	

5.3 C API		6.0 C++ API	
<b>FSPDF_LayerNode_Clear</b>		Replaced by class foxit::pdf::LayeNode	
<b>FSPDF_LayerNode_Init</b>		Replaced by class foxit::pdf::LayeNode	
<b>FSPDF_MarkedContent_AddItem</b>		foxit::pdf::graphics::MarkedContent::AddItem(const char* tag_name, objects::PDFDictionary* property_dict = NULL)	
<b>FSPDF_MarkedContent_CountItems</b>		foxit::pdf::graphics::MarkedContent::GetItemCount() const	
<b>FSPDF_MarkedContent_DeleteItem</b>		foxit::pdf::graphics::MarkedContent::RemoveItem(const char* tag_name)	
<b>FSPDF_MarkedContent_GetItemP aram</b>		foxit::pdf::graphics::MarkedContent::GetItemMCID(int index) const foxit::pdf::graphics::MarkedContent::GetItemPropertyDict(int index) const	
<b>FSPDF_MarkedContent_GetTagNa me</b>		foxit::pdf::graphics::MarkedContent::GetItemTagName(int index) const	
<b>FSPDF_MarkedContent_HasTag</b>		foxit::pdf::graphics::MarkedContent::HasTag(const char* tag_name)	
<b>FSPDF_Matrix_TransformDeviceTo Page</b>		Replaced by foxit::Matirx::TransformRect	
<b>FSPDF_Matrix_TransformPageToD evice</b>		Replaced by foxit::Matirx::TransformRect	
<b>FSPDF_Metadata_DeleteKey</b>		foxit::pdf::Metadata::RemoveCustomerKey(const wchar_t* key)	Only used to remove custom key
<b>FSPDF_Metadata_GetDateTime</b>		foxit::pdf::Metadata::GetCreationDateTime() const foxit::pdf::Metadata::GetModifiedDateTime() const	
<b>FSPDF_Metadata_GetKeys</b>		foxit::pdf::Metadata::GetCustomerKeys()	Only used to get custom key list
<b>FSPDF_Metadata_GetString</b>		foxit::pdf::Metadata::GetValues(const wchar_t* key) const	
<b>FSPDF_Metadata_GetStringArray</b>		foxit::pdf::Metadata::GetValues(const wchar_t* key) const	
<b>FSPDF_Metadata_SetDateTime</b>		foxit::pdf::Metadata::SetCreationDateTime(const DateTime& date_time) foxit::pdf::Metadata::SetModifiedDateTime(const DateTime& date_time)	
<b>FSPDF_Metadata_SetString</b>		foxit::pdf::Metadata::SetValues(const wchar_t* key, const WStringArray& value_array)	
<b>FSPDF_Metadata_SetStringArray</b>		foxit::pdf::Metadata::SetValues(const wchar_t* key, const WStringArray& value_array)	
<b>FSPDF_Object_CreateArray</b>		foxit::pdf::objects::PDFArray::Create()	
<b>FSPDF_Object_CreateBoolean</b>		foxit::pdf::objects::PDFObject::CreateFromBoolean(bool boolean_value)	
<b>FSPDF_Object_CreateDateTime</b>		foxit::pdf::objects::PDFObject::CreateFromDateTime(const DateTime& date_time)	
<b>FSPDF_Object_CreateDict</b>		foxit::pdf::objects::PDFDictionary::Create()	
<b>FSPDF_Object_CreateFloat</b>		foxit::pdf::objects::PDFObject::CreateFromFloat(float float_value)	
<b>FSPDF_Object_CreateInteger</b>		foxit::pdf::objects::PDFObject::CreateFromInteger(int integer_value)	
<b>FSPDF_Object_CreateMatrix</b>		foxit::pdf::objects::PDFArray::CreateFromMatrix(const Matrix& matrix)	
<b>FSPDF_Object_CreateRawByteStri ng</b>		N/A	
<b>FSPDF_Object_CreateRect</b>		foxit::pdf::objects::PDFArray::CreateFromRect(const RectF& rect)	
<b>FSPDF_Object_CreateReference</b>		N/A	
<b>FSPDF_Object_CreateReferenceWi thObjNum</b>		foxit::pdf::objects::PDFObject::CreateReference(Doc document, uint32 object_number)	
<b>FSPDF_Object_CreateStream</b>		foxit::pdf::objects::PDFStream::Create(PDFDictionary* dictionary)	
<b>FSPDF_Object_CreateUnicodeNam e</b>		foxit::pdf::objects::PDFObject::CreateFromName(const char* name)	



5.3 C API		6.0 C++ API	
<b>FSPDF_Object_CreateUnicodeString</b>		foxit::pdf::objects::PDFObject::CreateFromString(const wchar_t* string)	
<b>FSPDF_Object_GetBoolean</b>		foxit::pdf::objects::PDFObject::GetBoolean() const	
<b>FSPDF_Object_GetDateTime</b>		foxit::pdf::objects::PDFObject::GetDateTime() const	
<b>FSPDF_Object_GetFloat</b>		foxit::pdf::objects::PDFObject::GetFloat() const	
<b>FSPDF_Object_GetInteger</b>		foxit::pdf::objects::PDFObject::GetInteger() const	
<b>FSPDF_Object_GetMatrix</b>		foxit::pdf::objects::PDFObject::GetMatrix() const	
<b>FSPDF_Object_GetObjNum</b>		foxit::pdf::objects::PDFObject::GetObjNum() const	
<b>FSPDF_Object_GetRawByteString</b>		N/A	
<b>FSPDF_Object_GetRect</b>		foxit::pdf::objects::PDFObject::GetRect() const	
<b>FSPDF_Object_GetType</b>		foxit::pdf::objects::PDFObject::GetType() const	
<b>FSPDF_Object_GetUnicodeString</b>		foxit::pdf::objects::PDFObject::GetWideString() const	
<b>FSPDF_Object_Release</b>		foxit::pdf::objects::PDFObject::Release()	
<b>FSPDF_Page_CalcContentBBox</b>		foxit::pdf::Page::CalcContentBBox(CalcMarginMode mode)	
<b>FSPDF_Page_Clear</b>		foxit::pdf::Page::~~Page()	
<b>FSPDF_Page_CountActions</b>		foxit::pdf::actions::AdditionalAction::AdditionalAction(const Page& page) foxit::pdf::actions::AdditionalAction::GetAction(TriggerEvent trigger) // Count the sub-action of a specific action foxit::pdf::actions::Action::GetSubActionCount()	Every action may have sub-action, users can count them by yourself.
<b>FSPDF_Page_Create</b>		foxit::pdf::Doc::InsertPage(int index, float width, float height)	
<b>FSPDF_Page_Delete</b>		foxit::pdf::Doc::RemovePage(const Page& page)	
<b>FSPDF_Page_DoJavaScriptAction</b>		foxit::pdf::actions::AdditionalAction::AdditionalAction(const Page& page) foxit::pdf::actions::AdditionalAction::DoJSAction(TriggerEvent trigger)	
<b>FSPDF_Page_EstimateContentSize</b>		N/A	
<b>FSPDF_Page_Flatten</b>		foxit::pdf::Page::Flatten(bool for_display, uint32 options = Page::e_FlattenAll)	
<b>FSPDF_Page_GetAction</b>		foxit::pdf::actions::AdditionalAction::AdditionalAction(const Page& page) foxit::pdf::actions::AdditionalAction::GetAction(TriggerEvent trigger) // Get a specified sub-action if sub-action exists foxit::pdf::actions::Action::GetSubAction(int index)	
<b>FSPDF_Page_GetBox</b>		foxit::pdf::Page::GetBox(BoxType box_type)	
<b>FSPDF_Page_GetDict</b>		foxit::pdf::Page::GetDict() const	
<b>FSPDF_Page_GetIndex</b>		foxit::pdf::Page::GetIndex() const	
<b>FSPDF_Page_GetMatrix</b>		foxit::pdf::Page::GetDisplayMatrix(int left, int top, int width, int height, common::Rotation rotate)	
<b>FSPDF_Page_GetPageObjects</b>		Not Required.	class pdf::Pagefoxit:: extends class foxit::pdf::GraphicsObjects
<b>FSPDF_Page_GetRotation</b>		foxit::pdf::Page::GetRotation() const	
<b>FSPDF_Page_GetSize</b>		foxit::pdf::Page::GetHeight() const foxit::pdf::Page::GetWidth() const	
<b>FSPDF_Page_GetThumbnail</b>		foxit::pdf::Page::LoadThumbnail()	
<b>FSPDF_Page_HasTransparency</b>		foxit::pdf::Page::HasTransparency()	
<b>FSPDF_Page_InsertAction</b>		foxit::pdf::actions::AdditionalAction::AdditionalAction(const Page& page) foxit::pdf::actions::AdditionalAction::GetAction(TriggerEvent trigger) // Insert a sub-action foxit::pdf::actions::Action::InsertSubAction(int index, const Action& sub_action)	

5.3 C API		6.0 C++ API	
<b>FSPDF_Page_IsParsed</b>		foxit::pdf::Page::IsParsed() const	
<b>FSPDF_Page_LoadAnnots</b>		Not Required.	
<b>FSPDF_Page_RemoveAction</b>		foxit::pdf::actions::AdditionalAction::AdditionalAction(const Page& page) foxit::pdf::actions::AdditionalAction::GetAction(TriggerEvent trigger) // Remove a specified sub-action if sub-action exists foxit::pdf::actions::Action::RemoveSubAction(int index)  // or remove directly related action foxit::pdf::actions::AdditionalAction::RemoveAction(TriggerEvent trigger)	
<b>FSPDF_Page_RemoveAllActions</b>		foxit::pdf::actions::AdditionalAction::AdditionalAction(const Page& page) foxit::pdf::actions::AdditionalAction::RemoveAllActions()	
<b>FSPDF_Page_RemoveWatermarks</b>		foxit::pdf::Page::RemoveAllWatermarks()	
<b>FSPDF_Page_SetAction</b>		foxit::pdf::actions::AdditionalAction::AdditionalAction(const Page& page) foxit::pdf::actions::AdditionalAction::GetAction(TriggerEvent trigger) // Set a specified sub-action if sub-action exists foxit::pdf::actions::Action::SetSubAction(int index, const Action& sub_action)  // or set directly related action foxit::pdf::actions::AdditionalAction::SetAction(TriggerEvent trigger, const Action& action)	
<b>FSPDF_Page_SetBox</b>		foxit::pdf::Page::SetBox(BoxType box_type, const RectF& box)	
<b>FSPDF_Page_SetClipRect</b>		foxit::pdf::Page::SetClipRect(const RectF& clip_rect)	
<b>FSPDF_Page_SetIndex</b>		foxit::pdf::Doc::MovePageTo(const Page& page, int dest_index)	
<b>FSPDF_Page_SetRotation</b>		foxit::pdf::Page::SetRotation(common::Rotation rotate)	
<b>FSPDF_Page_SetSize</b>		foxit::pdf::Page::SetSize(float width, float height)	
<b>FSPDF_Page_SetThumbnail</b>		foxit::pdf::Page::SetThumbnail(const common::Bitmap& thumbnail)	
<b>FSPDF_Page_StartParse</b>		foxit::pdf::Page::StartParse(uint32 flags = Page::e_ParsePageNormal , common::PauseCallback* pause = NULL, bool is_reparse = false)	
<b>FSPDF_Page_Transform</b>		foxit::pdf::Page::Transform(const Matrix& matrix, bool need_transform_clip_path)	
<b>FSPDF_Page_UnloadAnnots</b>		Not Required.	
<b>FSPDF_PageLabels_CountRanges</b>		Not Required.	
<b>FSPDF_PageLabels_GetRangeInfo</b>		foxit::pdf::PageLabels::GetPageLabelStyle(int page_index) const foxit::pdf::PageLabels::GetPageLabelPrefix(int page_index) const foxit::pdf::PageLabels::GetPageLabelStart(int page_index) const	
<b>FSPDF_PageLabels_RemoveAll</b>		foxit::pdf::PageLabels::RemoveAll()	
<b>FSPDF_PageLabels_SetRangeInfo</b>		foxit::pdf::PageLabels::SetPageLabel(int page_index, Style label_style, int first_label_number, const WString& prefix_string = Wstring())	
<b>FSPDF_PageObject_AddClipPath</b>		foxit::pdf::graphics::GraphicsObject::AddClipPath(const common::Path& path, common::FillMode fill_mode)	
<b>FSPDF_PageObject_AddClipText</b>		foxit::pdf::graphics::GraphicsObject::AddClipTextObject(TextObject* textobject)	
<b>FSPDF_PageObject_ClearClips</b>		foxit::pdf::graphics::GraphicsObject::ClearClips()	
<b>FSPDF_PageObject_Clone</b>		foxit::pdf::graphics::GraphicsObject::Clone()	
<b>FSPDF_PageObject_CountClipPath</b>		foxit::pdf::graphics::GraphicsObject::GetClipPathCount() const	
<b>FSPDF_PageObject_CountClipText</b>		foxit::pdf::graphics::GraphicsObject::GetClipTextObjectCount()	

5.3 C API		6.0 C++ API	
<b>FSPDF_PageObject_DeleteClipPath</b>		foxit::pdf::graphics::GraphicsObject::RemoveClipPath(int index)	
<b>FSPDF_PageObject_DeleteClipText</b>		foxit::pdf::graphics::GraphicsObject::RemoveClipTextObject(int index)	
<b>FSPDF_PageObject_GetClipPath</b>		foxit::pdf::graphics::GraphicsObject::GetClipPath(int index) const foxit::pdf::graphics::GraphicsObject::GetClipPathFillMode(int index)	
<b>FSPDF_PageObject_GetClipRect</b>		foxit::pdf::graphics::GraphicsObject::GetClipRect() const	
<b>FSPDF_PageObject_GetClipText</b>		foxit::pdf::graphics::GraphicsObject::GetClipTextObject(int index) const	
<b>FSPDF_PageObject_GetColor</b>		foxit::pdf::graphics::GraphicsObject::GetStrokeColor() const foxit::pdf::graphics::GraphicsObject::GetFillColor() const	
<b>FSPDF_PageObject_GetGraphState</b>		foxit::pdf::graphics::GraphicsObject::GetGraphState() const	
<b>FSPDF_PageObject_GetLayers</b>		foxit::pdf::graphics::GraphicsObject::GetLayers(const LayerTree& layer_tree)	
<b>FSPDF_PageObject_GetMarkedContent</b>		foxit::pdf::graphics::GraphicsObject::GetMarkedContent() const	
<b>FSPDF_PageObject_GetMatrix</b>		foxit::pdf::graphics::GraphicsObject::GetMatrix() const	
<b>FSPDF_PageObject_GetRect</b>		foxit::pdf::graphics::GraphicsObject::GetRect() const	
<b>FSPDF_PageObject_GetType</b>		foxit::pdf::graphics::GraphicsObject::GetType() const	
<b>FSPDF_PageObject_HasTransparency</b>		foxit::pdf::graphics::GraphicsObject::HasTransparency()	
<b>FSPDF_PageObject_Release</b>		foxit::pdf::graphics::GraphicsObject::Release()	
<b>FSPDF_PageObject_SetClipRect</b>		foxit::pdf::graphics::GraphicsObject::SetClipRect(const RectF& clip_rect)	
<b>FSPDF_PageObject_SetColor</b>		foxit::pdf::graphics::GraphicsObject::SetStrokeColor(ARGB color) foxit::pdf::graphics::GraphicsObject::SetFillColor(ARGB color)	
<b>FSPDF_PageObject_SetGraphState</b>		foxit::pdf::graphics::GraphicsObject::SetGraphState(const common::GraphState& graph_state)	
<b>FSPDF_PageObject_SetMatrix</b>		foxit::pdf::graphics::GraphicsObject::SetMatrix(const Matrix& matrix)	
<b>FSPDF_PageObject_Transform</b>		foxit::pdf::graphics::GraphicsObject::Transform(const Matrix& matrix, bool need_transform_clippath)	
<b>FSPDF_PageObjects_CountObjects</b>		Not Required.	
<b>FSPDF_PageObjects_EnumerateObject</b>		foxit::pdf::GraphicsObjects::GetFirstGraphicsObjectPosition(graphics::GraphicsObject::Type filter = graphics::GraphicsObject::e_TypeAll) const  foxit::pdf::GraphicsObjects::GetNextGraphicsObjectPosition(POSITION position, graphics::GraphicsObject::Type filter = graphics::GraphicsObject::e_TypeAll) const  foxit::pdf::GraphicsObjects::GetLastGraphicsObjectPosition(graphics::GraphicsObject::Type filter = graphics::GraphicsObject::e_TypeAll) const  foxit::pdf::GraphicsObjects::GetPrevGraphicsObjectPosition(POSITION position, graphics::GraphicsObject::Type filter = graphics::GraphicsObject::e_TypeAll) const  foxit::pdf::GraphicsObjects::GetGraphicsObject(POSITION position) const	
<b>FSPDF_PageObjects_GenerateContents</b>		foxit::pdf::GraphicsObjects::GenerateContent()	
<b>FSPDF_PageObjects_GetObject</b>		Not Required.	

5.3 C API		6.0 C++ API	
<b>FSPDF_PageObjects_GetObjectAtPos</b>		foxit::pdf::Page::GetGraphicsObjectAtPoint(const PointF& point, float tolerance, graphics::GraphicsObject::Type filter = graphics::GraphicsObject::e_TypeAll) const	
<b>FSPDF_PageObjects_GetObjectIndex</b>		Not Required.	
<b>FSPDF_PageObjects_GetObjectListAtPos</b>		foxit::pdf::Page::GetGraphicsObjectsAtPoint(const PointF& point, float tolerance, graphics::GraphicsObject::Type filter = graphics::GraphicsObject::e_TypeAll)	
<b>FSPDF_PageObjects_InsertObject</b>		foxit::pdf::GraphicsObjects::InsertGraphicsObject(POSITION position_insert_after, graphics::GraphicsObject* graphics_object)	
<b>FSPDF_PageObjects_RemoveObject</b>		foxit::pdf::GraphicsObjects::RemoveGraphicsObject(graphics::GraphicsObject* graphics_object)	
<b>FSPDF_PathObject_ConvertFromTextObject</b>		foxit::pdf::graphics::PathObject::CreateFromTextObject(const Page& page, TextObject* text_obj)	
<b>FSPDF_PathObject_Create</b>		foxit::pdf::graphics::PathObject::Create()	
<b>FSPDF_PathObject_GetFillMode</b>		foxit::pdf::graphics::PathObject::GetFillMode() const	
<b>FSPDF_PathObject_GetPathData</b>		foxit::pdf::graphics::PathObject::GetPathData() const	
<b>FSPDF_PathObject_GetStrokeState</b>		foxit::pdf::graphics::PathObject::GetStrokeState() const	
<b>FSPDF_PathObject_SetFillMode</b>		foxit::pdf::graphics::PathObject::SetFillMode(common::FillMode fill_mode)	
<b>FSPDF_PathObject_SetPathData</b>		foxit::pdf::graphics::PathObject::SetPathData(const common::Path& path_data)	
<b>FSPDF_PathObject_SetStrokeState</b>		foxit::pdf::graphics::PathObject::SetStrokeState(bool is_stroke)	
<b>FSPDF_Redaction_Apply</b>		foxit::addon::Redaction::Apply()	
<b>FSPDF_Redaction_Mark</b>		foxit::addon::Redaction::MarkRedactAnnot(const pdf::Page& page, const RectFArray& rects)	
<b>FSPDF_Reference_GetReferObject</b>		foxit::pdf::objects::PDFObject::GetDirectObject()	
<b>FSPDF_ReflowPage_Create</b>		foxit::pdf::ReflowPage::ReflowPage(const Page& page)	
<b>FSPDF_ReflowPage_GetContentSize</b>		foxit::pdf::ReflowPage::GetContentWidth() const foxit::pdf::ReflowPage::GetContentHeight() const	
<b>FSPDF_ReflowPage_GetFocusData</b>		foxit::pdf::ReflowPage::GetFocusData(const Matrix& matrix, const PointF& point) const	
<b>FSPDF_ReflowPage_GetFocusPosition</b>		foxit::pdf::ReflowPage::GetFocusPosition(const Matrix& matrix, const char* focus_data) const	
<b>FSPDF_ReflowPage_GetMatrix</b>		foxit::pdf::ReflowPage::GetDisplayMatrix(float offset_x, float offset_y)	
<b>FSPDF_ReflowPage_Release</b>		foxit::pdf::ReflowPage::~ReflowPage()	
<b>FSPDF_ReflowPage_SetLineSpace</b>		foxit::pdf::ReflowPage::SetLineSpace(float line_space)	
<b>FSPDF_ReflowPage_SetSize</b>		foxit::pdf::ReflowPage::SetScreenSize(float width, float height)	
<b>FSPDF_ReflowPage_SetTopSpace</b>		foxit::pdf::ReflowPage::SetTopSpace(float top_space)	
<b>FSPDF_ReflowPage_StartParse</b>		foxit::pdf::ReflowPage::SetParseFlags(uint32 flags) foxit::pdf::ReflowPage::StartParse(common::PauseCallback* pause = NULL)	
<b>FSPDF_RenderContext_Create</b>		Not Required.	
<b>FSPDF_RenderContext_DrawSignature</b>		N/A	
<b>FSPDF_RenderContext_Release</b>		Not Required.	

5.3 C API		6.0 C++ API	
<b>FSPDF_RenderContext_SetFlags</b>		foxit::common::Renderer::SetTransformAnnotIcon(bool is_to_transform_annot_icon) foxit::common::Renderer::SetClearType(bool is_clear_type) foxit::common::Renderer::SetPrintTextAsGraphic(bool is_to_print_text_graphic) foxit::common::Renderer::SetPrintTextAsImage(bool is_to_print_text_image) foxit::common::Renderer::SetForceDownSample(bool is_to_force_down_sample) foxit::common::Renderer::SetRenderPathThinLine(bool is_render_path_thin_line) foxit::common::Renderer::SetRenderTextAntiAliasing(bool is_render_text_antialiasing) foxit::common::Renderer::SetRenderPathAntiAliasing(bool is_render_path_antialiasing) foxit::common::Renderer::SetRenderImageAntiAliasing(bool is_render_image_antialiasing)	
<b>FSPDF_RenderContext_SetHalftoneLimit</b>		foxit::common::Renderer::SetForceHalftone(bool is_to_force_halftone)	Only set whether uses halftone limit, and external does not need to set limit value.
<b>FSPDF_RenderContext_SetLayerContext</b>		foxit::common::Renderer::SetLayerContext(const pdf::LayerContext& layer_context)	
<b>FSPDF_RenderContext_SetMatrix</b>		Not Required.	To be a parameter of render function.
<b>FSPDF_RenderContext_SetOption</b>		foxit::common::Renderer::SetColorMode(ColorMode color_mode) foxit::common::Renderer::SetMappingModeColors(ARGB background_color, ARGB foreground_color)	
<b>FSPDF_RenderContext_StartAnnotations</b>		foxit::common::Renderer::RenderAnnot(const pdf::annots::Annot& annot, const Matrix& matrix)	
<b>FSPDF_RenderContext_StartFormControls</b>		foxit::common::Renderer::SetRenderFormField(bool is_render_formfield) foxit::common::Renderer::StartRender(const pdf::Page& page, const Matrix& matrix, PauseCallback* pause = NULL)	
<b>FSPDF_RenderContext_StartPage</b>		foxit::common::Renderer::StartRender(const pdf::Page& page, const Matrix& matrix, PauseCallback* pause = NULL)	
<b>FSPDF_RenderContext_StartPageAnnots</b>		foxit::common::Renderer::SetRenderContentFlags(uint32 render_content_flags) foxit::common::Renderer::StartRender(const pdf::Page& page, const Matrix& matrix, PauseCallback* pause = NULL)	
<b>FSPDF_RenderContext_StartPageFormControls</b>		foxit::common::Renderer::SetRenderFormField(bool is_render_formfield) foxit::common::Renderer::StartRender(const pdf::Page& page, const Matrix& matrix, PauseCallback* pause = NULL)	
<b>FSPDF_RenderContext_StartPageSignatures</b>		foxit::common::Renderer::SetRenderSignature(bool is_render_signature) foxit::common::Renderer::StartRender(const pdf::Page& page, const Matrix& matrix, PauseCallback* pause = NULL)	
<b>FSPDF_RenderContext_StartReflowPage</b>		foxit::common::Renderer::StartRenderReflowPage(const pdf::ReflowPage& reflow_page, const Matrix& matrix, PauseCallback* pause = NULL)	
<b>FSPDF_Security_CheckPassword</b>		foxit::pdf::Doc::CheckPassword(const String& password) foxit::pdf::Doc::CheckPassword(const WString& password)	
<b>FSPDF_Security_GetFoxitDRMEncryptionParamValue</b>		foxit::pdf::DRMSecurityHandler::GetDRMValue(const char* key)	
<b>FSPDF_Security_RegisterHandler</b>		foxit::common::Library::RegisterSecurityCallback(const char* filter, pdf::SecurityCallback* callback)	
<b>FSPDF_Security_SetCertificateHandler</b>		foxit::common::Library::RegisterSecurityCallback(const char* filter, pdf::SecurityCallback* callback)	

5.3 C API		6.0 C++ API	
<b>FSPDF_Security_SetFoxitDRMEncryptionParamValue</b>		foxit::pdf::DRMSecurityHandler::SetDRMValue(const char* key, const WString& value)	
<b>FSPDF_Security_SetFoxitDRMHandler</b>		foxit::common::Library::RegisterSecurityCallback(const char* filter, pdf::SecurityCallback* callback)	
<b>FSPDF_Security_StartCertificateEncryption</b>		foxit::pdf::CertificateSecurityHandler::CertificateSecurityHandler() foxit::pdf::CertificateSecurityHandler::Initialize foxit::pdf::Doc::SetSecurityHandler(const SecurityHandler& handler) foxit::pdf::Doc::SaveAs or StartSaveAs	
<b>FSPDF_Security_StartCustomEncryption</b>		foxit::pdf::CustomSecurityHandler::CustomSecurityHandler() foxit::pdf::CustomSecurityHandler::Initialize(const CustomEncryptData& encrypt_data, const CustomSecurityCallback * callback, const char* encrypt_info = "") foxit::pdf::Doc::SetSecurityHandler(const SecurityHandler& handler) foxit::pdf::Doc::SaveAs() or StartSaveAs()	
<b>FSPDF_Security_StartFoxitDRMEncryption</b>		foxit::pdf::DRMSecurityHandler::DRMSecurityHandler() foxit::pdf::DRMSecurityHandler::Initialize(const DRMEncryptData& encrypt_data, const char* file_id, const String& initial_key) foxit::pdf::Doc::SetSecurityHandler(const SecurityHandler& handler) foxit::pdf::Doc::SaveAs() or StartSaveAs()	
<b>FSPDF_Security_StartPasswordEncryption</b>		foxit::pdf::StdSecurityHandler::StdSecurityHandler() foxit::pdf::StdSecurityHandler::Initialize(const StdEncryptData& encrypt_data, const String& user_password, const String& owner_password) foxit::pdf::Doc::SetSecurityHandler(const SecurityHandler& handler) foxit::pdf::Doc::SaveAs() or StartSaveAs()	
<b>FSPDF_Security_StartRMSEncryption</b>		foxit::pdf::RMSSecurityHandler::RMSSecurityHandler() foxit::pdf::RMSSecurityHandler::Initialize(const RMSEncryptData& encrypt_data, const RMSSecurityCallback* callback) foxit::pdf::Doc::SetSecurityHandler(const SecurityHandler& handler) foxit::pdf::Doc::SaveAs() or StartSaveAs()	
<b>FSPDF_Security_UnregisterHandler</b>		foxit::common::Library::UnregisterSecurityCallback(const char* filter)	
<b>FSPDF_Security_VerifyFoxitDRMEncryptionParams</b>		foxit::pdf::DRMSecurityHandler::VerifyEncryptionParams()	
<b>FSPDF_ShadingObject_GetShading</b>		foxit::pdf::graphics::ShadingObject::GetPDFObject() const	
<b>FSPDF_Signature_Add</b>		foxit::pdf::Page::AddSignature(const RectF& rect)	
<b>FSPDF_Signature_ClearData</b>		foxit::pdf::Signature::ClearSignedData()	
<b>FSPDF_Signature_CountCerts</b>		foxit::pdf::Signature::GetCertCount() const	
<b>FSPDF_Signature_GetAppearanceFlags</b>		foxit::pdf::Signature::GetAppearanceFlags()	
<b>FSPDF_Signature_GetAtPos</b>		Not Required.	Get form field or widget annot
<b>FSPDF_Signature_GetBitmap</b>		foxit::pdf::Signature::GetBitmap()	
<b>FSPDF_Signature_GetCert</b>		foxit::pdf::Signature::GetCert(int32 index) const	
<b>FSPDF_Signature_GetContactInfo</b>		foxit::pdf::Signature::GetKeyValue(KeyName key)	
<b>FSPDF_Signature_GetDateTime</b>		foxit::pdf::Signature::GetSignTime()	
<b>FSPDF_Signature_GetDistinguishedName</b>		foxit::pdf::Signature::GetKeyValue(KeyName key)	
<b>FSPDF_Signature_GetDoc</b>		foxit::pdf::Signature::GetDocument()	

5.3 C API		6.0 C++ API	
<b>FSPDF_Signature_GetFilter</b>		foxit::pdf::Signature::GetFilter()	
<b>FSPDF_Signature_GetFlags</b>		foxit::pdf::interform::Field::GetControl(0) foxit::pdf::interform::GetWidget() foxit::pdf::annots::Annot::GetFlags()	
<b>FSPDF_Signature_GetLocation</b>		foxit::pdf::Signature::GetKeyValue(KeyName key)	
<b>FSPDF_Signature_GetPageIndex</b>		foxit::pdf::interform::Field::GetControl(0) foxit::pdf::interform::GetWidget() foxit::pdf::annots::Annot::GetPage() foxit::pdf::Page::GetIndex()	
<b>FSPDF_Signature_GetReason</b>		foxit::pdf::Signature::GetKeyValue(KeyName key)	
<b>FSPDF_Signature_GetSigner</b>		foxit::pdf::Signature::GetKeyValue(KeyName key)	
<b>FSPDF_Signature_GetState</b>		foxit::pdf::Signature::GetState()	
<b>FSPDF_Signature_GetSubFilter</b>		foxit::pdf::Signature::GetSubFilter()	
<b>FSPDF_Signature_GetText</b>		foxit::pdf::Signature::GetKeyValue(KeyName key)	
<b>FSPDF_Signature_InitValue</b>		Not Required.	
<b>FSPDF_Signature_IsSigned</b>		foxit::pdf::Signature::IsSigned()	
<b>FSPDF_Signature_RegisterHandler</b>		foxit::common::Library::RegisterSignatureCallback(const char* filter, const char* sub_filter, pdf::SignatureCallback* signature_callback)	
<b>FSPDF_Signature_Remove</b>		foxit::pdf::Doc::RemoveSignature(const Signature& signature)	
<b>FSPDF_Signature_ResetAppearance</b>		Not Required.	Generated when signing signature
<b>FSPDF_Signature_SetAppearanceFlags</b>		foxit::pdf::Signature::SetAppearanceFlags(uint32 appearance_flags)	
<b>FSPDF_Signature_SetBitmap</b>		foxit::pdf::Signature::SetBitmap(const common::Bitmap& bitmap)	
<b>FSPDF_Signature_SetCertChain</b>		foxit::pdf::Signature::SetCertChain(const WStringArray& cert_chain)	
<b>FSPDF_Signature_SetContactInfo</b>		foxit::pdf::Signature::SetKeyValue(KeyName key, const wchar_t* value)	
<b>FSPDF_Signature_SetDateTime</b>		foxit::pdf::Signature::SetSignTime(const DateTime& sign_time)	
<b>FSPDF_Signature_SetDefaultContentsLength</b>		foxit::pdf::Signature::SetDefaultContentsLength(uint32 default_length)	
<b>FSPDF_Signature_SetDistinguishedName</b>		foxit::pdf::Signature::SetKeyValue(KeyName key, const wchar_t* value)	
<b>FSPDF_Signature_SetFilter</b>		foxit::pdf::Signature::SetFilter(const char* filter)	
<b>FSPDF_Signature_SetFlags</b>		foxit::pdf::interform::Field::GetControl(0) foxit::pdf::interform::GetWidget() foxit::pdf::annots::Annot::SetFlags()	
<b>FSPDF_Signature_SetIconProvider</b>		N/A	
<b>FSPDF_Signature_SetImage</b>		foxit::pdf::Signature::SetImage(const common::Image& image, int frame_index) foxit::pdf::Signature::SetImage(const char* file_path, int frame_index) foxit::pdf::Signature::SetImage(const wchar_t* file_path, int frame_index)	
<b>FSPDF_Signature_SetLocation</b>		foxit::pdf::Signature::SetKeyValue(KeyName key, const wchar_t* value)	
<b>FSPDF_Signature_SetReason</b>		foxit::pdf::Signature::SetKeyValue(KeyName key, const wchar_t* value)	
<b>FSPDF_Signature_SetSigner</b>		foxit::pdf::Signature::SetKeyValue(KeyName key, const wchar_t* value)	
<b>FSPDF_Signature_SetSubFilter</b>		foxit::pdf::Signature::SetSubFilter(const char* sub_filter)	
<b>FSPDF_Signature_SetText</b>		foxit::pdf::Signature::SetKeyValue(KeyName key, const wchar_t* value)	

5.3 C API		6.0 C++ API	
<b>FSPDF_Signature_StartSign</b>		foxit::pdf::Signature::StartSign(const wchar_t* cert_path, const WString& cert_password, DigestAlgorithm digest_algorithm, const char* save_path, const void* client_data = NULL, common::PauseCallback* pause = NULL) foxit::pdf::Signature::StartSign(const wchar_t* cert_path, const WString& cert_password, DigestAlgorithm digest_algorithm, const wchar_t* save_path, const void* client_data = NULL, common::PauseCallback* pause = NULL)	
<b>FSPDF_Signature_StartVerify</b>		foxit::pdf::Signature::StartVerify(const void* client_data = NULL, common::PauseCallback* pause = NULL)	
<b>FSPDF_Stream_ExportData</b>		foxit::pdf::objects::PDFStream::ExportData(bool is_raw_data)	
<b>FSPDF_Stream_GetData</b>		foxit::pdf::objects::PDFStream::GetDataSize(bool is_raw_data) Foxit::pdf::objects::PDFStream::GetData(bool is_raw_data, size_t size, void* out_buffer)	
<b>FSPDF_Stream_GetDict</b>		foxit::pdf::objects::PDFStream::GetDictionary()	
<b>FSPDF_Stream_ImportData</b>		foxit::pdf::objects::PDFStream::ImportData(common::file::ReaderCallback* file_read, Filter stream_filter)	
<b>FSPDF_Stream_SetData</b>		foxit::pdf::objects::PDFStream::SetData(const void* buffer, size_t size)	
<b>FSPDF_TextLink_CountLinks</b>		foxit::pdf::PageTextLinks::GetTextLinkCount()	
<b>FSPDF_TextLink_GetLink</b>		foxit::pdf::PageTextLinks::GetTextLink(int index)	
<b>FSPDF_TextLink_GetSelection</b>		After getting TextLink, foxit::pdf::TextLink::GetURI() foxit::pdf::TextLink::GetStartCharIndex() foxit::pdf::TextLink::GetEndCharIndex() foxit::pdf::TextLink::GetRects()	
<b>FSPDF_TextLink_Release</b>		foxit::pdf::TextLink::~~TextLink()	foxit::pdf::TextLink::GetURI()
<b>FSPDF_TextObject_Create</b>		foxit::pdf::graphics::TextObject::Create()	foxit::pdf::TextLink::GetStartCharIndex()
<b>FSPDF_TextObject_GetTextState</b>		foxit::pdf::graphics::TextObject::GetTextState(const Page& page) const	foxit::pdf::TextLink::GetEndCharIndex()
<b>FSPDF_TextObject_GetUnicodeString</b>		foxit::pdf::graphics::TextObject::GetText() const	foxit::pdf::TextLink::GetRects()
<b>FSPDF_TextObject_SetTextState</b>		foxit::pdf::graphics::TextObject::SetTextState(const Page& page, const TextState& text_state, bool is_italic, int weight)	
<b>FSPDF_TextObject_SetUnicodeString</b>		foxit::pdf::graphics::TextObject::SetText(const wchar_t* text)	
<b>FSPDF_TextPage_CountChars</b>		foxit::pdf::TextPage::GetCharCount() const	
<b>FSPDF_TextPage_ExportToFile</b>		N/A	
<b>FSPDF_TextPage_ExtractLinks</b>		foxit::pdf::PageTextLinks::PageTextLinks(const TextPage& page)	
<b>FSPDF_TextPage_GetCharIndexAtPos</b>		foxit::pdf::TextPage::GetIndexAtPos(float x, float y, float tolerance) const	
<b>FSPDF_TextPage_GetCharInfo</b>		N/A	
<b>FSPDF_TextPage_GetChars</b>		foxit::pdf::TextPage::GetChars(int start = 0, int count = -1) const	
<b>FSPDF_TextPage_GetNextCharIndexByDirection</b>	弃用 deprecated	Not Required.	
<b>FSPDF_TextPage_GetUnicode</b>		N/A	
<b>FSPDF_TextPage_Load</b>		foxit::pdf::TextPage::TextPage(const Page& page, int flags = foxit::pdf::TextPage::e_ParseTextNormal)	



5.3 C API		6.0 C++ API	
<b>FSPDF_TextPage_LoadWithOptions</b>		foxit::pdf::TextPage::TextPage(const Page& page, int flags = foxit::pdf::TextPage::e_ParseTextNormal)	
<b>FSPDF_TextPage_Release</b>		foxit::pdf::TextPage::~~TextPage()	
<b>FSPDF_TextPage_SelectByRange</b>		Not Required.	
<b>FSPDF_TextPage_SelectByRectangle</b>		Not Required.	
<b>FSPDF_TextPage_StartSearch</b>		foxit::pdf::TextSearch::TextSearch(const TextPage& text_page)	
<b>FSPDF_TextSearch_FindNext</b>		foxit::pdf::TextSearch::FindNext()	
<b>FSPDF_TextSearch_FindPrev</b>		foxit::pdf::TextSearch::FindPrev()	
<b>FSPDF_TextSearch_GetSelection</b>		foxit::pdf::TextSearch::GetMatchRects() const foxit::pdf::TextSearch::GetMatchPageIndex() const foxit::pdf::TextSearch::GetMatchSentence() foxit::pdf::TextSearch::GetMatchSentenceStartIndex() foxit::pdf::TextSearch::GetMatchStartCharIndex() const foxit::pdf::TextSearch::GetMatchEndCharIndex() const	
<b>FSPDF_TextSearch_Release</b>		foxit::pdf::TextSearch::~~TextSearch()	
<b>FSPDF_TextSelection_CountPieces</b>		foxit::pdf::TextPage::GetTextRectCount(int start = 0, int count = -1)	
<b>FSPDF_TextSelection_GetBBox</b>		Not Required.	
<b>FSPDF_TextSelection_GetChars</b>		foxit::pdf::TextPage::GetChars(int start = 0, int count = -1) const	
<b>FSPDF_TextSelection_GetLayerArray</b>		Not Required.	
<b>FSPDF_TextSelection_GetPieceCharRange</b>		Not Required.	
<b>FSPDF_TextSelection_GetPieceRect</b>		foxit::pdf::TextPage::GetTextRect(int rect_index) const	
<b>FSPDF_TextSelection_GetPieceRotation</b>		foxit::pdf::TextPage::GetBaselineRotation(int rect_index)	
<b>FSPDF_TextSelection_GetVisibleSelection</b>		Not Required.	
<b>FSPDF_TextSelection_Release</b>		Not Required.	
<b>FSPDF_ViewerPref_GetAreaType</b>		foxit::pdf::DocViewerPrefs::GetViewArea() foxit::pdf::DocViewerPrefs::GetViewClip() foxit::pdf::DocViewerPrefs::GetPrintArea() foxit::pdf::DocViewerPrefs::GetPrintClip()	
<b>FSPDF_ViewerPref_GetNonFullScreenPageMode</b>		foxit::pdf::DocViewerPrefs::GetNonFullScreenPageMode() const	
<b>FSPDF_ViewerPref_GetPrintCopies</b>		foxit::pdf::DocViewerPrefs::GetPrintCopies() const	
<b>FSPDF_ViewerPref_GetPrintRanges</b>		foxit::pdf::DocViewerPrefs::GetPrintRange() const	
<b>FSPDF_ViewerPref_GetPrintScale</b>		foxit::pdf::DocViewerPrefs::GetPrintScale() const	
<b>FSPDF_ViewerPref_GetReadingDirection</b>		foxit::pdf::DocViewerPrefs::GetReadingDirection() const	
<b>FSPDF_ViewerPref_GetUIVisibility</b>		foxit::pdf::DocViewerPrefs::GetUIDisplayStatus(ViewerPref pref) const	
<b>FSPDF_ViewerPref_SetAreaType</b>		foxit::pdf::DocViewerPrefs::SetViewArea(Page::BoxType box_type) foxit::pdf::DocViewerPrefs::SetViewClip(Page::BoxType box_type)	

5.3 C API		6.0 C++ API	
		foxit::pdf::DocViewerPrefs::SetPrintArea(Page::BoxType box_type) foxit::pdf::DocViewerPrefs::SetPrintClip(Page::BoxType box_type)	
<b>FSPDF_ViewerPref_SetNonFullScreenPageMode</b>		foxit::pdf::DocViewerPrefs::SetNonFullScreenPageMode(Doc::DisplayMode display_mode)	
<b>FSPDF_ViewerPref_SetPrintCopies</b>		foxit::pdf::DocViewerPrefs::SetPrintCopies(int print_copies)	
<b>FSPDF_ViewerPref_SetPrintRanges</b>		foxit::pdf::DocViewerPrefs::SetPrintRange(const common::Range& page_range)	
<b>FSPDF_ViewerPref_SetPrintScale</b>		foxit::pdf::DocViewerPrefs::SetPrintScale(PrintScale print_scale)	
<b>FSPDF_ViewerPref_SetReadingDirection</b>		foxit::pdf::DocViewerPrefs::SetReadingDirection(bool is_left_to_right)	
<b>FSPDF_ViewerPref_SetUIVisibility</b>		foxit::pdf::DocViewerPrefs::SetUIDisplayStatus(ViewerPref pref, bool value)	
<b>FSPDF_Watermark_CreateFromBitmap</b>		foxit::pdf::Watermark::Watermark(const Doc& document, const common::Bitmap& bitmap, const WatermarkSettings& settings)	
<b>FSPDF_Watermark_CreateFromImage</b>		foxit::pdf::Watermark::Watermark(const Doc& document, const common::Image& image, int frame_index, const WatermarkSettings& settings)	
<b>FSPDF_Watermark_CreateFromPage</b>		foxit::pdf::Watermark::Watermark(const Doc& document, const Page& page, const WatermarkSettings& settings)	
<b>FSPDF_Watermark_CreateFromText</b>		foxit::pdf::Watermark::Watermark(const Doc& document, const wchar_t* text, const WatermarkTextProperties& properties, const WatermarkSettings& settings)	
<b>FSPDF_Watermark_GetCount</b>		N/A	Currently only provide foxit::pdf::Page::HasWatermark() function to confirm whether the page contains watermark.
<b>FSPDF_Watermark_GetSize</b>		foxit::pdf::Watermark::GetWidth() const foxit::pdf::Watermark::GetHeight() const	
<b>FSPDF_Watermark_InsertToPage</b>		foxit::pdf::Watermark::InsertToPage(Page page)	
<b>FSPDF_Watermark_Release</b>		foxit::pdf::Watermark::~~Watermark()	
<b>FSPDF_Watermark_Remove</b>		N/A	currently only provide foxit::pdf::Page::RemoveAllWatermarks() function to remove all of the watermarks.
<b>FSPDF_WrapperData_Clear</b>		Replaced by class foxit::pdf::WrapperData	
<b>FSPDF_WrapperData_Init</b>		Replaced by class foxit::pdf::WrapperData	