

dEXIF

dexif

Version 1.0 • Proposed

Date/Time Generated:

06.08.2017 16:13:12

Author:

andi

EA Repository : C:\Data\Pascal\komponenten\dexif\doc\dExif.EAP

Table of Contents

dexif	3
dexif diagram	7
Note	8
dEXIF	8
dIPTC	9
msData	9
dEXIF	9
dEXIF diagram	10
TGpsFormat	12
TImageInfo	12
TImgData	18
tEndInd	22
tSection	23
dIPTC	25
dIPTC diagram	25
TIPTCdata	26
TTagEntry	29
msData	31
msData diagram	31
TmsInfo	31
dEXIF	32

dexif

Package in package 'Domain Model'

Programmer's Guide to dEXIF and dIPTC

Image Metadata Manipulators Written in Delphi.

Gerry McGuire

mcguirez@hotmail.com

Copyright 2001-2006 - All Rights Reserved

Theory of Operation:

An EXIF header is a multilevel binary structure with the following characteristics: (Several details are glossed over here - a more complete specification can be found at http://www.pima.net/standards/iso/tc42/wg18/WG18_POW.htm)

- * The image file is divided into sections. These sections are separated by a marker byte followed by a section type byte.
- * One of these sections is the "EXIF" section. This section is used by the capturing device (camera) to record the various conditions and settings in effect at the time of acquisition.
- * Another of these sections is the "IPTC" section. IPTC stands for the "International Press Telecommunications Council" and this section holds metadata relevant for publishing an image. This section contains data such as "Caption", "Byline", "Urgency" etc. Photoshop can read and write to this area and is available under "Image Info..." on the file menu.
- * Another one of these sections is the "Comments" section. It basically holds up to 64k of data. I believe there may be more than one Comments section although dEXIF only handles the first one at this time.
- * An EXIF section consists of a directory structure. There is only one valid EXIF section per image. This section may also contain several sub-sections which each contain their own directory.
- * An EXIF directory consists of entries that consist of
 - a) a 16 bit field identifier (Tag)
 - b) a 16 bit type indicator
 - c) a 32 bit component count (used to determine the data length)
 - d) If the data is 4 bytes in size or less, it is contained here otherwise a 32 bit pointer to the data is contained here.
- * One of the EXIF entries is the "MakerNote" entry. This information varies by the manufacturer. Typically this is also a directory structure that can be interpreted in the same fashion as the EXIF directory.
- * Typical jpegs have a JFIF section which is a brief 16 byte header.

* An EXIF section may be substituted for a JFIF section but there must be one or the other. If a JFIF section exists it must be first, an EXIF entry must also be first unless there is also a JFIF section in which case the EXIF section must be second. (Anyone remember Monty Python's holy hand grenade incantation?)

* A jpeg, using Borland's jpeg unit, will not be altered unless the compress method is called. That is, no degradation of the image is caused by continually reading and writing via this library.

* If a thumbnail is present, a pointer to the thumbnail block occupies the 32 bytes immediately after the EXIF IFD0 directory. In some formats (NIKON) there may be information following the thumbnail block but still in the EXIF segment.

* The routines read the EXIF information from the file, copy it to a buffer, and create the TAG list from the buffer.

File -----> EXIF Buffer -----> Tag list -----> User

Therefore changes to the Tag list will not be written to a file. There are two methods on the ExifObj (WriteThruInt and WriteThruString) that alter the EXIF buffer and changes can be written to an image file.

Sample Code:

dEXIF handles the major sections by parsing the initial image header and putting each into it's own buffer. This is currently accomplished by the dEXIF unit which also processes the EXIF data. The dIPTC processes the IPTC section - surprise~!

An IPTC segment and a Comments segment can be added to an arbitrary jpeg file, but an EXIF structure cannot be added. This is because EXIF contains mostly camera setting etc, that cannot be meaningfully generated. An EXIF structure can be copied over, presumably from a file that was the root source of the targeted image file.

Sample Code [EXIF Section]:

The following examples assume this declaration:

```
Var ImgData: TImgData;
Begin
  ImgData := TImgData.Create;
```

Open a file, extract the simple EXIF string summary:

```
// The following returns true if successful. Will parse both
// jpgs and tiff files.
if ImgData.ProcessFile('filename.jpg') then
  if ImgData.HasExif then
    MyString := ImgData.ExifObj.toString();
```

Open a file, extract a more detailed EXIF data summary:

```
// One string with embedded (CR/LF) breaks.
if ImgData.ProcessFile('filename.jpg') then
```

```

if ImgData.HasExif then
  MyString := ImgData.ExifObj.toLongString();

```

Open a file, extract all EXIF data, interpreted (decoded) with all possible detail.

```

DexifDecode := true; // true by default - translates literals
                    // into something human-readable
ImgData.TraceLevel := 1; // set detail level for trace string
// One string with embedded (CR/LF) breaks.
if ImgData.ProcessFile('filename.jpg') then
  if ImgData.HasExif then
    // Parse this to extract specific data
    MyTraceString := ImgData.ExifObj.TraceString

```

Some of the More common fields are parsed already and placed in the TImageInfo class. ImageInfo is predeclared and is used by the global read routines. (This will change - I'm working on it!)

With ImgData do

```

if ProcessFile('filename.jpg') and HasExif then // Read in file
begin
  // These first 3 are read from the OS
  MyFileName := ExifObj.FileName;
  FileDate := ExifObj.FileDateTime; // tDateTime;
  FileSize := ExifObj.FileSize; // longint;
  // These are extracted from the image
  Make := ExifObj.CameraMake; // string[32];
  Model := ExifObj.CameraModel; // string[40];
  DateTime := ExifObj.DateTime; // string[20];
  Height := ExifObj.Height
  Width := ExifObj.Width; // integer
  YPos := ExifObj.HPosn; // integer
  XPos := ExifObj.WPosn; // integer
  IsColor := ExifObj.IsColor; // integer;
  Process := ExifObj.Process; // integer;
  FlashUsed := ExifObj.FlashUsed; // integer;
  // These are derived (computed) values
  FocalLength := ExifObj.FocalLength; // real;
  ExposureTime := ExifObj.ExposureTime; // real;
  ApertureFNumber := ExifObj.ApertureFNumber; // real;
  Distance := ExifObj.Distance; // real;
  CCDWidth := ExifObj.CCDWidth; // real;
  Comments := ExifObj.Comments; // string;
end;

```

Open a file, manipulate the image and save to a new file adjusting the EXIF size for the new file:

```

// var MyImage:TJpegImage;
if ImgData.ProcessFile('FileName.jpg') then
begin
  MyImage.ReadFromFile('FileName.jpg');
  // manipulate image here
  ImgData.WriteEXIF(MyImage,'NewFile.jpg');
end;

```

Open a file and save to a new file removing all metadata:

```

// var MyImage:TJpegImage;
if ImgData.ProcessFile('FileName.jpg') then

```

```
begin
  ImgData.ClearSections(); // Get rid of all MetaData
  ImgData.WriteEXIF('NewFile.jpg'); // When only one name is specified,
end;                      // The last file read is used as source.
```

Sample Code [Comments Section]:

The comments section can hold upto 64k bytes of data.
Don't confuse it with the UserComments tag of the EXIF segment, which is fixed by the camera and is much shorter.

Open a jpeg file and extract the comments section into a string. note that the comment section can contain binary data - it's not necessarily a string:

```
ImgData.ProcessFile('filename.jpg'); // calls the EXIF parsing routine
if ImgData.CommentSection > 0 then // there was a Comment Section
  MyString := ImgData.GetCommentSegment();
```

Open a file, do nothing other than add a tstringlist to the comment section and save:

```
ImgData.ProcessFile('filename.jpg'); // calls the EXIF parsing routine
// The following call will overwrite
ImgData.MakeCommentSegment(MyString); // or add a comment section
```

Sample Code [IPTC Section]:

Read IPTC strings from a jpeg file:

```
// We're not interested to deal with EXIF data
StringList := ImgData.IPTCObj.ReadFileStrings('filename.jpg');
```

Read IPTC data into dIPTC for further manipulation, check the copyright tag, add it if it doesn't exist and save the file:

```
// normally use a WITH block but this IS an example:
if ImgData.IPTCObj.ReadFile('filename.jpg') then
begin // file parsed OK - and there was actual data
  if ImgData.IPTCObj.GetTag('Copyright') = '' then
  begin
    ImgData.IPTCObj.AddTag('Copyright','Copyright 2004 - Gerry McGuire');
    ImgData.IPTCObj.WriteFile(); // defaults to last ReadFile value
  end
end;
```

Adding IPTC data to a generated image which does not yet exist as a file: (typically you'd use a With ImgData.IPTCObj... statement)

```
ImgData.IPTCObj.Reset; // clear out old values
ImgData.IPTCObj.AddTag('Copyright','Copyright 2004 - Gerry McGuire');
ImgData.IPTCObj.AddTag('Keywords','Software, Sample');
// To add a keyword without removing earlier values
ImgData.IPTCObj.AppendToTag('Keywords','Another');
ImgData.IPTCObj.AddTag('OriginatingProgram','dEXIF Version 1.2c');
ImgData.IPTCObj.WriteFile('newfile.jpg',NewJpgImage); // Last argument is TJpegImage
```

Read in a file and save any IPTC data as XML:

```
ImgData.IPTCObj.ReadFile('filename.jpg'); // as an alternative, you can
if ImgData.IPTCObj.HasData then           // check for data after the ReadFile
    ImgData.IPTCObj.IPTCArrayToXML('filename.xml');
```

Note that the EXIF section has a similar function and that the `ImgData` object has a method (`MetaDataToXML()`) which returns a stringlist that contains the XML of both segments.

Read all of the currently defined tags and their labels into a stringgrid. The data array is `TIPTCdata`'s default value.

```
// var grid: tstringgrid; i:integer;
grid.colcount := 2;
grid.rowcount := ImgData.IPTCObj.count;
for i := 0 to ImgData.IPTCObj.Count-1 do
begin
    grid.cells[0,i] := ImgData.IPTCObj[i].desc; // get label value
    grid.cells[1,i] := ImgData.IPTCObj[i].data; // get actual data
end;
```

Write a file containing all the user tag descriptions:

```
ImgData.IPTCObj.IPTCWriteTransFile( 'translate.txt');
```

Read in file containing all the user tag descriptions (may be used for translating UI into a non-English language). Perform this in a form create method or before any images are read in.

```
ImgData.IPTCObj.IPTCReadTransFile( 'translate.txt');
```

See the `IPTCView` application bundled with this source code for a more detailed example than those presented above.

dexif

Version 1.0 Phase 1.0 Proposed

Andreas Frieß created on 04.08.2017. Last modified 04.08.2017

dexif diagram

Class diagram in package 'dexif'

dexif

Version 1.0

Andreas Frieß created on 04.08.2017. Last modified 04.08.2017

dEXIF is Copyright © 2001 - 2006 Gerry McGuire. All rights reserved.
See the copyright details specified in Copyright.txt

Written and tested in Delphi 7.0
now ported to Lazarus/Freepascal 2017

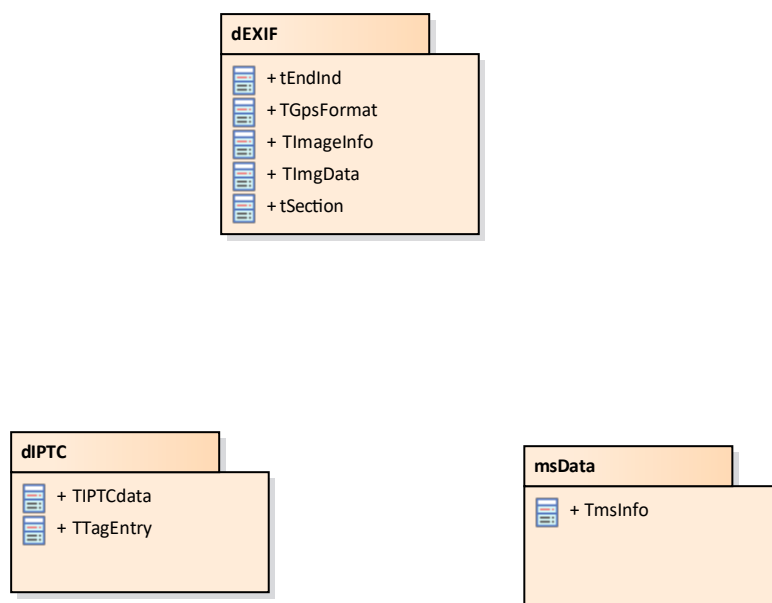


Figure 1: dexif

Note

Note in package 'dexif'

dEXIF is Copyright © 2001 - 2006 Gerry McGuire. All rights reserved.
See the copyright details specified in Copyright.txt

Written and tested in Delphi 7.0
now ported to Lazarus/Freepascal 2017

Note
Version 1.0 Phase 1.0 Proposed
Andreas Frieß created on 04.08.2017. Last modified 04.08.2017
Extends

dEXIF

Package in package 'dexif'

dEXIF
Version 1.0 Phase 1.0 Proposed
Andreas Frieß created on 04.08.2017. Last modified 04.08.2017

ASSOCIATED FILES



Local File. C:\data\Pascal\dexif\dEXIF.pas

[Size of 118 k.]

diPTC

Package in package 'dexif'

diPTC

Version 1.0 Phase 1.0 Proposed

Andreas Frieß created on 04.08.2017. Last modified 04.08.2017

ASSOCIATED FILES



Local File. C:\data\Pascal\dexif\diPTC.pas

[Size of 26 k.]

msData

Package in package 'dexif'

msData

Version 1.0 Phase 1.0 Proposed

Andreas Frieß created on 04.08.2017. Last modified 04.08.2017

ASSOCIATED FILES



Local File. C:\data\Pascal\dexif\msData.pas

[Size of 35 k.]

dEXIF

Package in package 'dexif'

dEXIF

Version 1.0 Phase 1.0 Proposed

Andreas Frieß created on 04.08.2017. Last modified 04.08.2017

ASSOCIATED FILES



Local File. C:\data\Pascal\dexif\dEXIF.pas

[Size is 118 k]

dEXIF diagram

Class diagram in package 'dEXIF'

dEXIF

Version 1.0

Andreas Frieß created on 04.08.2017. Last modified 04.08.2017

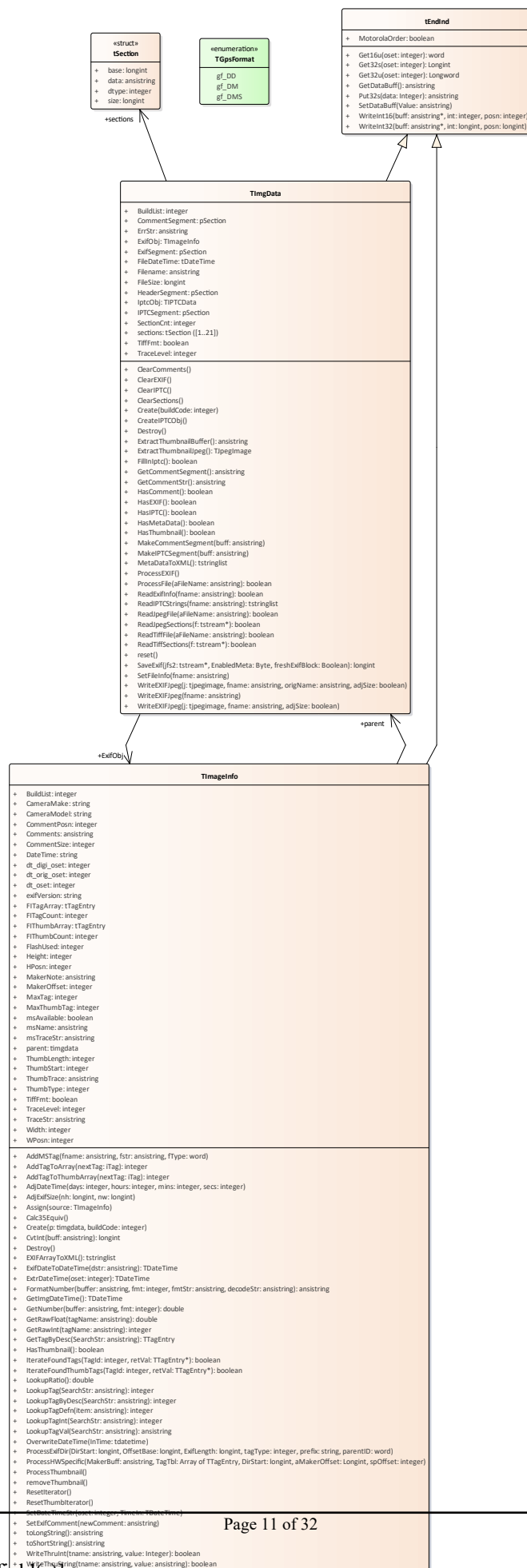


Figure 2: dEXIF

TGpsFormat




Enumeration «enumeration» in package 'dEXIF'

TTagTableArray = array of TTagEntry;

TGpsFormat

Version 1.0 Phase 1.0 Proposed

Andreas Frieß created on 04.08.2017. Last modified 04.08.2017

ATTRIBUTES	
 gf_DD : Public	[Is static False. Containment is Not Specified.]
 gf_DM : Public	[Is static False. Containment is Not Specified.]
 gf_DMS : Public	[Is static False. Containment is Not Specified.]

TImageInfo

Class in package 'dEXIF'


TImageInfo




TImageInfo

Version 1.0 Phase 1.0 Proposed

Andreas Frieß created on 04.08.2017. Last modified 04.08.2017














Extends tEndInd

OUTGOING STRUCTURAL RELATIONSHIPS	
 Generalization from TImageInfo to tEndInd	[Direction is 'Source -> Destination'.]

ATTRIBUTES	
 BuildList : integer Public	[Is static False. Containment is By Value.]
 CameraMake : string Public	[Is static False. Containment is By Value.]
 CameraModel : string Public	[Is static False. Containment is By Value.]






ATTRIBUTES	
 CommentPosn : integer Public	[Is static False. Containment is By Value.]
 Comments : ansistring Public	[Is static False. Containment is By Value.]
 CommentSize : integer Public	[Is static False. Containment is By Value.]
 DateTime : string Public	[Is static False. Containment is By Value.]
 dt_digi_aset : integer Public	[Is static False. Containment is By Value.]
 dt_orig_aset : integer Public	[Is static False. Containment is By Value.]
 dt_aset : integer Public DateTime tag locations	[Is static False. Containment is By Value.]
 exifVersion : string Public	[Is static False. Containment is By Value.]
 FITagArray : tTagEntry Public	[Is static False. Containment is By Value.]
 FITagCount : integer Public	[Is static False. Containment is By Value.]
 FIThumbArray : tTagEntry Public	[Is static False. Containment is By Value.]
 FIThumbCount : integer Public	[Is static False. Containment is By Value.]
 FlashUsed : integer Public	[Is static False. Containment is By Value.]
 Height : integer Public	[Is static False. Containment is By Value.]
 HPosn : integer Public	[Is static False. Containment is By Value.]

ATTRIBUTES	
 MakerNote : ansistring Public	[Is static False. Containment is By Value.]
 MakerOffset : integer Public	[Is static False. Containment is By Value.]
 MaxTag : integer Public	[Is static False. Containment is By Value.]
 MaxThumbTag : integer Public	[Is static False. Containment is By Value.]
 msAvailable : boolean Public	[Is static False. Containment is By Value.]
 msName : ansistring Public	[Is static False. Containment is By Value.]
 msTraceStr : ansistring Public	[Is static False. Containment is By Value.]
 parent : timgdata Public	[Is static False. Containment is By Value.]
 ThumbLength : integer Public	[Is static False. Containment is By Value.]
 ThumbStart : integer Public	[Is static False. Containment is By Value.]
 ThumbTrace : ansistring Public Add support for thumbnail	[Is static False. Containment is By Value.]
 ThumbType : integer Public	[Is static False. Containment is By Value.]
 TiffFmt : boolean Public	[Is static False. Containment is By Value.]
 TraceLevel : integer Public Added the following elements to make the structure a little more code-friendly	[Is static False. Containment is By Value.]
 TraceStr : ansistring Public	[Is static False. Containment is By Value.]

ATTRIBUTES	
 Width : integer Public	[Is static False. Containment is By Value.]
 WPosn : integer Public	[Is static False. Containment is By Value.]
ASSOCIATIONS	
 Association (direction: Source -> Destination) Source: Public (Class) TImageInfo	Target: Public FITagArray (Class) TTagEntry «struct»
 Association (direction: Source -> Destination) Source: Public (Class) TImageInfo	Target: Public FIThumbArray (Class) TTagEntry «struct»
 Association (direction: Source -> Destination) Source: Public (Class) TImageInfo	Target: Public parent (Class) TImgData
 Association (direction: Source -> Destination) Source: Public (Class) TImgData	Target: Public ExifObj (Class) TImageInfo
 Association (direction: Source -> Destination) Source: Public (Class) TmsInfo	Target: Public IMGparent (Class) TImageInfo
OPERATIONS	
 AddMSTag (fname : ansistring , fstr : ansistring , fType : word) : Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
 AddTagToArray (nextTag : iTag) : integer Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
 AddTagToThumbArray (nextTag : iTag) : integer Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
 AdjDateTime (days : integer , hours : integer , mins : integer , secs : integer) : Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
 AdjExifSize (nh : longint , nw : longint) : Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
 Assign (source : TImageInfo) : Public	

OPERATIONS	
	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ Calc35Equiv () : Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ Create (p : timgdata , buildCode : integer) : Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ CvtInt (buff : ansistring) : longint Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ Destroy () : Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ EXIFArrayToXML () : tstringlist Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ ExifDateToDateTime (dstr : ansistring) : TDateTime Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ ExtrDateTime (oset : integer) : TDateTime Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ FormatNumber (buffer : ansistring , fmt : integer , fmtStr : ansistring , decodeStr : ansistring) : ansistring Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ GetImgDateTime () : TDateTime Public	
The following functions manage the date	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ GetNumber (buffer : ansistring , fmt : integer) : double Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ GetRawFloat (tagName : ansistring) : double Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ GetRawInt (tagName : ansistring) : integer Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ GetTagByDesc (SearchStr : ansistring) : TTagEntry Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ HasThumbnail () : boolean Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ IterateFoundTags (TagId : integer , retVal : TTagEntry) : boolean Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]


OPERATIONS
<p>◆ IterateFoundThumbTags (TagId : integer , retVal : TTagEntry) : boolean Public [Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]</p>
<p>◆ LookupRatio () : double Public [Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]</p>
<p>◆ LookupTag (SearchStr : ansistring) : integer Public [Is static False. Is abstract True. Is return array False. Is query False. Is synchronized False.]</p>
<p>◆ LookupTagByDesc (SearchStr : ansistring) : integer Public [Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]</p>
<p>◆ LookupTagDefn (item : ansistring) : integer Public [Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]</p>
<p>◆ LookupTagInt (SearchStr : ansistring) : integer Public [Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]</p>
<p>◆ LookupTagVal (SearchStr : ansistring) : ansistring Public [Is static False. Is abstract True. Is return array False. Is query False. Is synchronized False.]</p>
<p>◆ OverwriteDateTime (InTime : tdatetime) : Public [Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]</p>
<p>◆ ProcessExifDir (DirStart : longint , OffsetBase : longint , ExifLength : longint , tagType : integer , prefix : string , parentID : word) : Public [Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]</p>
<p>◆ ProcessHWSpecific (MakerBuff : ansistring , TagTbl : Array of TTagEntry , DirStart : longint , aMakerOffset : Longint , spOffset : integer) : Public Contains embedded CR/LFs [Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]</p>
<p>◆ ProcessThumbnail () : Public [Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]</p>
<p>◆ removeThumbnail () : Public [Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]</p>
<p>◆ ResetIterator () : Public [Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]</p>
<p>◆ ResetThumbIterator () : Public [Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]</p>
<p>◆ SetDateTimeStr (oset : integer , TimeIn : TDateTime) : Public</p>




OPERATIONS	
	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
 SetExifComment (newComment : ansistring) : Public [Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]	
 toLongString () : ansistring Public Summerizes in a single line [Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]	
 toShortString () : ansistring Public The following functions format this structure into a string [Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]	
 WriteThruInt (tname : ansistring , value : Integer) : boolean Public [Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]	
 WriteThruString (tname : ansistring , value : ansistring) : boolean Public [Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]	



TImgData

Class in package 'dEXIF'

TImgData
Version 1.0 Phase 1.0 Proposed
Andreas Frieß created on 04.08.2017. Last modified 04.08.2017
Extends tEndInd

OUTGOING STRUCTURAL RELATIONSHIPS	
 Generalization from TImgData to tEndInd [Direction is 'Source -> Destination'.]	

ATTRIBUTES	
 BuildList : integer Public [Is static False. Containment is By Value.]	
 CommentSegment : pSection Public [Is static False. Containment is By Value.]	
 ErrStr : ansistring Public [Is static False. Containment is By Value.]	
 ExifObj : TImageInfo Public [Is static False. Containment is By Value.]	

ATTRIBUTES	
 ExifSegment : pSection Public	[Is static False. Containment is By Value.]
 FileDateTime : tDateTime Public	[Is static False. Containment is By Value.]
 Filename : ansistring Public	[Is static False. Containment is By Value.]
 FileSize : longint Public	[Is static False. Containment is By Value.]
 HeaderSegment : pSection Public	[Is static False. Containment is By Value.]
 IptcObj : TIPTCData Public	[Is static False. Containment is By Value.]
 IPTCSegment : pSection Public	[Is static False. Containment is By Value.]
 SectionCnt : integer Public	[Is static False. Containment is By Value.]
 sections : tSection Public One per image object	[Is static False. Containment is By Value.]
 TiffFmt : boolean Public	[Is static False. Containment is By Value.]
 TraceLevel : integer Public	[Is static False. Containment is By Value.]

ASSOCIATIONS	
 Association (direction: Source -> Destination) Source: Public (Class) TImgData	Target: Public sections (Class) tSection «struct»
 Association (direction: Source -> Destination) Source: Public (Class) TImgData	Target: Public IptcObj (Class) TIPTCdata
 Association (direction: Source -> Destination) Source: Public (Class) TImgData	Target: Public ExifObj (Class) TImageInfo

ASSOCIATIONS

 Association (direction: Source -> Destination)

Source: Public (Class) TImageInfo

Target: Public parent (Class) TImgData

OPERATIONS

 ClearComments () : Public
[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]

 ClearEXIF () : Public
[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]

 ClearIPTC () : Public
[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]

 ClearSections () : Public
[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]

 Create (buildCode : integer) : Public
[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]

 CreateIPTCObj () : Public
[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]

 Destroy () : Public
[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]

 ExtractThumbnailBuffer () : ansistring Public
[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]


 ExtractThumbnailJpeg () : TjpegImage Public
[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]

 FillInIptc () : boolean Public
[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]


 GetCommentSegment () : ansistring Public
[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]

 GetCommentStr () : ansistring Public
[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]

 HasComment () : boolean Public
[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]

 HasEXIF () : boolean Public

OPERATIONS	
	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ HasIPTC () : boolean Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ HasMetaData () : boolean Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ HasThumbnail () : boolean Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ MakeCommentSegment (buff : ansistring) : Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ MakeIPTCSegment (buff : ansistring) : Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ MetaDataToXML () : tstringlist Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ ProcessEXIF () : Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ ProcessFile (aFileName : ansistring) : boolean Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ ReadExifInfo (fname : ansistring) : boolean Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ ReadIPTCStrings (fname : ansistring) : tstringlist Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ ReadJpegFile (aFileName : ansistring) : boolean Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ ReadJpegSections (f : tstream) : boolean Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ ReadTiffFile (aFileName : ansistring) : boolean Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ ReadTiffSections (f : tstream) : boolean Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ reset () : Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]

OPERATIONS	
 SaveExif (jfs2 : tstream , EnabledMeta : Byte , freshExifBlock : Boolean) : longint Public [Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]	
 SetFileInfo (fname : ansistring) : Public [Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]	
 WriteEXIFJpeg (j : tjpegimage , fname : ansistring , origName : ansistring , adjSize : boolean) : Public Properties: overload = true [Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]	
 WriteEXIFJpeg (fname : ansistring) : Public Properties: overload = true [Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]	
 WriteEXIFJpeg (j : tjpegimage , fname : ansistring , adjSize : boolean) : Public Properties: overload = true [Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]	

tEndInd





Class in package 'dEXIF'

tEndInd

tEndInd

Version 1.0 Phase 1.0 Proposed

Andreas Frieß created on 04.08.2017. Last modified 04.08.2017

INCOMING STRUCTURAL RELATIONSHIPS	
 Generalization from TImageInfo to tEndInd [Direction is 'Source -> Destination'.]	
 Generalization from TImgData to tEndInd [Direction is 'Source -> Destination'.]	
ATTRIBUTES	
 MotorolaOrder : boolean Public [Is static False. Containment is By Value.]	
OPERATIONS	
 Get16u (oset : integer) : word Public	

OPERATIONS	
	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ Get32s (oSet : integer) : Longint Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ Get32u (oSet : integer) : Longword Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ GetDataBuff () : ansistring Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ Put32s (data : Integer) : ansistring Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ SetDataBuff (Value : ansistring) : Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ WriteInt16 (buff : ansistring , int : integer , posn : integer) : Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ WriteInt32 (buff : ansistring , int : longint , posn : longint) : Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]

tSection

Class «struct» in package 'dEXIF'

TInfoData

tSection

Version 1.0 Phase 1.0 Proposed

Andreas Frieß created on 04.08.2017. Last modified 04.08.2017

ATTRIBUTES	
◆ base : longint Public	[Is static False. Containment is By Value.]
◆ data : ansistring Public	[Is static False. Containment is By Value.]
◆ dtype : integer Public	[Is static False. Containment is By Value.]
◆ size : longint Public	[Is static False. Containment is By Value.]

ASSOCIATIONS

Association (direction: Source -> Destination)

Source: Public (Class) TImgData

Target: Public sections (Class) tSection «struct»

diPTC

Package in package 'dexif'

diPTC

Version 1.0 Phase 1.0 Proposed

Andreas Frieß created on 04.08.2017. Last modified 04.08.2017

ASSOCIATED FILES



Local File. C:\data\Pascal\dexif\diPTC.pas

[Size is 26 k]

diPTC diagram

Class diagram in package 'diPTC'

diPTC

Version 1.0

Andreas Frieß created on 04.08.2017. Last modified 04.08.2017

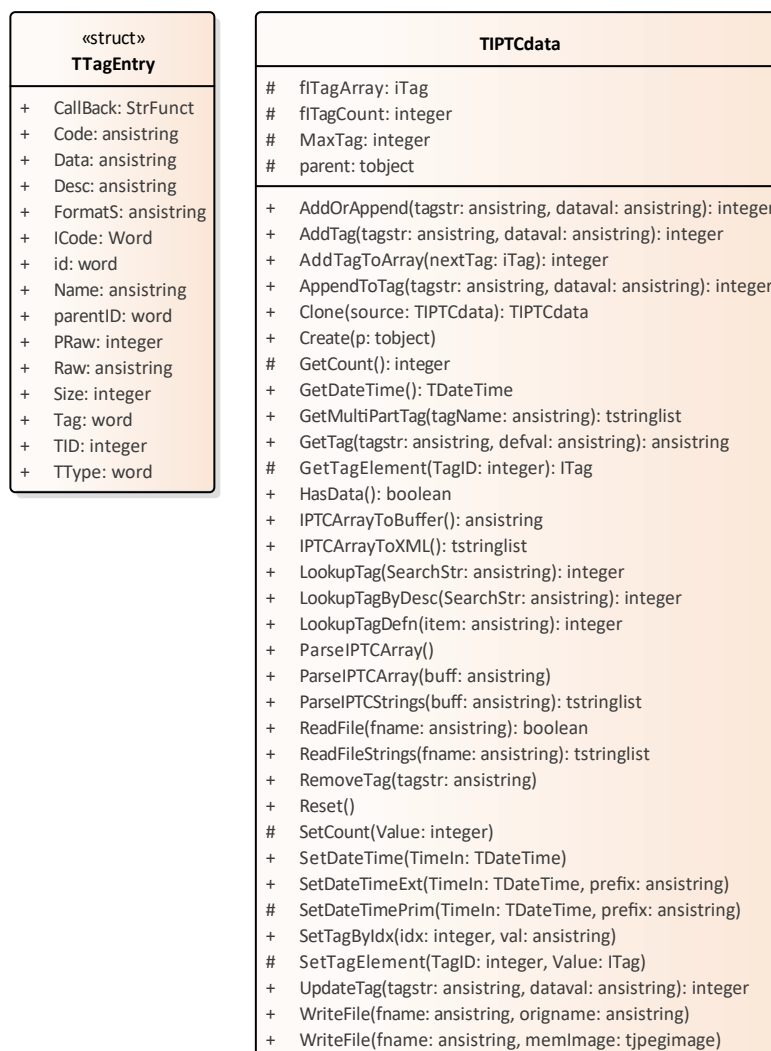


Figure 3: dIPTC












TIPTCdata

Class in package 'dIPTC'

TIPTCdata

Version 1.0 Phase 1.0 Proposed

Andreas Frieß created on 04.08.2017. Last modified 04.08.2017

ATTRIBUTES	
 flTagArray : iTag Protected	[Is static False. Containment is By Value.]
 flTagCount : integer Protected	[Is static False. Containment is By Value.]
 MaxTag : integer Protected	[Is static False. Containment is By Value.]
 parent : tobject Protected	[Is static False. Containment is By Value.]
ASSOCIATIONS	
 Association (direction: Source -> Destination) Source: Public (Class) TImgData Target: Public IptcObj (Class) TIPTCdata	
OPERATIONS	
 AddOrAppend (tagstr : ansistring , dataval : ansistring) : integer Public	[Is static False. Is abstract True. Is return array False. Is query False. Is synchronized False.]
 AddTag (tagstr : ansistring , dataval : ansistring) : integer Public	[Is static False. Is abstract True. Is return array False. Is query False. Is synchronized False.]
 AddTagToArray (nextTag : iTag) : integer Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
 AppendToTag (tagstr : ansistring , dataval : ansistring) : integer Public	[Is static False. Is abstract True. Is return array False. Is query False. Is synchronized False.]
 Clone (source : TIPTCdata) : TIPTCdata Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
 Create (p : tobject) : Public	

OPERATIONS	
Filename :ansistring;	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ GetCount () : integer Protected	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ GetDateTime () : TDateTime Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ GetMultiPartTag (tagName : ansistring) : tstringlist Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ GetTag (tagstr : ansistring , defval : ansistring) : ansistring Public	[Is static False. Is abstract True. Is return array False. Is query False. Is synchronized False.]
◆ GetTagElement (TagID : integer) : ITag Protected	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ HasData () : boolean Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ IPTCArrayToBuffer () : ansistring Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ IPTCArrayToXML () : tstringlist Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ LookupTag (SearchStr : ansistring) : integer Public	[Is static False. Is abstract True. Is return array False. Is query False. Is synchronized False.]
◆ LookupTagByDesc (SearchStr : ansistring) : integer Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ LookupTagDefn (item : ansistring) : integer Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ ParseIPTCArray () : Public Properties: overload = true	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ ParseIPTCArray (buff : ansistring) : Public Properties: overload = true	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ ParseIPTCStrings (buff : ansistring) : tstringlist Public	

OPERATIONS	
	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ ReadFile (fname : ansistring) : boolean Public	[Is static False. Is abstract True. Is return array False. Is query False. Is synchronized False.]
◆ ReadFileStrings (fname : ansistring) : tstringlist Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ RemoveTag (tagstr : ansistring) : Public	[Is static False. Is abstract True. Is return array False. Is query False. Is synchronized False.]
◆ Reset () : Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ SetCount (Value : integer) : Protected	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ SetDateTime (TimeIn : TDateTime) : Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ SetDateTimeExt (TimeIn : TDateTime , prefix : ansistring) : Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ SetDateTimePrim (TimeIn : TDateTime , prefix : ansistring) : Protected	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ SetTagByIdx (idx : integer , val : ansistring) : Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ SetTagElement (TagID : integer , Value : ITag) : Protected	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ UpdateTag (tagstr : ansistring , dataval : ansistring) : integer Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ WriteFile (fname : ansistring , origname : ansistring) : Public Properties: overload = true	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ WriteFile (fname : ansistring , memImage : tjpegimage) : Public Properties: overload = true	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]

TTagEntry


Class «struct» in package 'dIPTC'

TTagEntry

Version 1.0 Phase 1.0 Proposed


Andreas Frieß created on 04.08.2017. Last modified 04.08.2017

ATTRIBUTES		
 CallBack : StrFunc Public	formatting string	[Is static False. Containment is By Value.]
 Code : ansistring Public	decode capability	[Is static False. Containment is By Value.]
 Data : ansistring Public	display value	[Is static False. Containment is By Value.]
 Desc : ansistring Public	translatable	[Is static False. Containment is By Value.]
 FormatS : ansistring Public	Format string	[Is static False. Containment is By Value.]
 ICode : Word Public	iptc code	[Is static False. Containment is By Value.]
 id : word Public	msta - used for exif-parent-child-structure	[Is static False. Containment is By Value.]
 Name : ansistring Public	searchable	[Is static False. Containment is By Value.]
 parentID : word Public	msta - used for exif-parent-child-structure	[Is static False. Containment is By Value.]

ATTRIBUTES PRaw : integer Public


pointer to unprocessed

[Is static False. Containment is By Value.]

 Raw : ansistring Public


unprocessed value

[Is static False. Containment is By Value.]

 Size : integer Public


used by ITPC module

[Is static False. Containment is By Value.]

 Tag : word Public


primary key

[Is static False. Containment is By Value.]

 TID : integer Public

TagTableID - EXIF use

[Is static False. Containment is By Value.]

 TType : word Public

tag type

[Is static False. Containment is By Value.]

ASSOCIATIONS Association (direction: Source -> Destination)

Source: Public (Class) TImageInfo

Target: Public FITagArray (Class) TTagEntry
«struct» Association (direction: Source -> Destination)

Source: Public (Class) TImageInfo

Target: Public FIThumbArray (Class) TTagEntry
«struct»

msData

Package in package 'dexif'

msData

Version 1.0 Phase 1.0 Proposed

Andreas Frieß created on 04.08.2017. Last modified 04.08.2017

ASSOCIATED FILES



Local File. C:\data\Pascal\dexif\msData.pas

[Size is 35 k]

msData diagram

Class diagram in package 'msData'

msData

Version 1.0

Andreas Frieß created on 04.08.2017. Last modified 04.08.2017

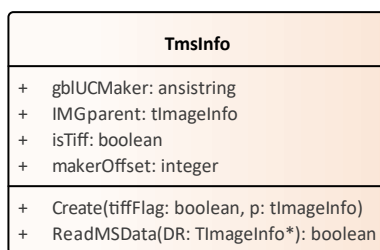


Figure 4: msData

TmsInfo

Class in package 'msData'

TmsInfo

Version 1.0 Phase 1.0 Proposed

Andreas Frieß created on 04.08.2017. Last modified 04.08.2017

ATTRIBUTES



gblUCMaker : ansistring Public

[Is static False. Containment is By Value.]



IMGparent : tImageInfo Public

[Is static False. Containment is By Value.]




isTiff : boolean Public

manufactorer specific

[Is static False. Containment is By Value.]

ATTRIBUTES

 makerOffset : integer Public

[Is static False. Containment is By Value.]


ASSOCIATIONS

 Association (direction: Source -> Destination)

Source: Public (Class) TmsInfo

Target: Public IMGparent (Class) TImageInfo

OPERATIONS

 Create (tiffFlag : boolean , p : TImageInfo) : Public

[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]

 ReadMSData (DR : TImageInfo) : boolean Public

[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]

dEXIF

UMLDiagram in package 'dexif'

dEXIF

Version 1.0 Phase 1.0 Proposed

Andreas Frieß created on 04.08.2017. Last modified 04.08.2017

Extends