-

Image File Format Description

LSM 5/7

Release 6.0 (ZEN 2010)

Documentation 6.0.5

Changes / additions to the document 6.0 from September 28, 2010 are marked in yellow
Additions to the document v 6.0.2 from Oct 5, 2010 are marketd in cyan
Addition to v 6.0.4 from Jan 6, 2011 is the Chapter 14 (Application Tags / KS data)

June 22, 2011

Carl Zeiss MicroImaging GmbH
BioSciences

For information about .lsm file format please contact Zeiss Product Management in Jena
(Olaf Selchow (selchow@zeiss.de)

Contents

Release notes

In Release 3.5 the LSM 5 family has been extended with the LSM 5 LIVE device. LSM 510, LSM 5 PASCAL and LSM 5 LIVE are controlled by the AIM-Software, which stores the images in ".lsm"-files with the same format for all these devices. The information about the acquisition in the ".lsm"-file has been extended by new entries to support the LSM 5 LIVE as well. Since the LSM 5 LIVE system is a fast acquisition device, the term "Real-time Scanner" or RT-Scanner is used by most of the elements of data structures and names of the constants to identify information relevant for the LSM 5 LIVE system only.

In addition, the file reader and writer have completely been rewritten in Release 3.5 of the AIM-Software, with the intent to keep the format compatible. It was expected, that image file readers based on the Release 3.2 file format description are still able to read the files. In the meantime two problems have been identified which caused file readers to reject some of the newer ".lsm" files:

1. The three entries RECORDING_ENTRY_PLANE_WIDTH, RECORDING_ENTRY_PLANE_HEIGHT and RECORDING_ENTRY_VOLUME_DEPTH of the acquisition information tree are not present in files written by the Release 3.5 version of the AIM Software. They have been considered to contain redundant information. In the Release 4.0 version the entries are written again. However, a reader should be aware that there are files were these entries are not present.
2. The new file writers of Release 3.5 and 4.0 use a different order of the sections in the file. Together with a deviation of the meaning of the TIF_STRIPBYTECOUNTS tag for compressed files this caused a problem for some readers for compressed files. In ".lsm" format one strip per image plane and channel is generated. In contrast to the TIFF specification the STRIPBYTECOUNTS tag contains the size of the uncompressed data, not the compressed data. This way a reader, which uses the STRIPBYTECOUNTS tag to read the file content might try to read data behind the end of the file. For compatibility reasons we decided not to change the meaning of the STRIPBYTECOUNTS tag in ".lsm" files. To overcome the problem it is recommended to check against the file size when a compressed image block is read. The AIM-Program itself uses the more sophisticated technique by checking the strip offsets and file size to determine the size of the compressed data in the file.

In Release 4.5 (ZEN 2007) the usage of an image database has been replaced by direct handling of ".lsm" files. There have been minor extensions which did not affect the general structure of the files.

In Release 5.0 (ZEN 2008) the LSM 7 family has been introduced with the LSM 710 and extended by the LSM 700 device in Release 5.5 (ZEN 2009). Several minor extensions have been necessary for the new devices. Again the basic file structure has not changed.

In Release 6.0 (ZEN 2010), adjustments to new hardware have been made. No change of the basic file structure.

1. LSM 5/7 Image Files (*.lsm)

The LSM 5/7 file format is basically a multi image TIFF-format with some restrictions.

The major restriction results from the fact that the devices are able to generate more than 3 image channels. In the TIFF-format specification 1-channel images (monochrome or palette images) and 3-channel images (RGB images) are described in detail. For the LSM 5/7 format the „SAMPLESPERPIXEL“ tag is set to the number of channels, which can be a value up to 1024 for release 5.0. Most of the readers do not expect „SAMPLESPERPIXEL“-entries other than 1 or 3. There are several commercial programs that are able to read LSM-Files for 3-channel and 1-channel acquisition with the TIFF import filter, but without the additional information in the file.

The different versions of the writer in the LSM programs had several deviations form the TIFF standard. These deviations can be treated to be bugs. In two cases the bugs have been fixed. In the other two cases a fix would have prevented older versions of the LSM program to read the newer files. A reader for all variants of existing LSM files should therefore consider the following four descriptions of the deviations:

- LSM files generated with version 1.6 and earlier have TIFF-tags, which have not been stored in ascending order.

- LSM files generated with version 1.6 and earlier have palette entries were the lower 8-bits have been used instead of the upper 8 bits. The reader can check the upper 8 bits. When at least one bit is set, the palette intensity data are in the upper 8 bits, otherwise in the lower 8 bits.

- In contrast to the TIFF standard the STRIPBYTECOUNTS tag contains the number of bytes for the uncompressed data.

  In a first method to read compressed data, the uncompressed data size of all channels of an image plane can be determined form the STRIPBYTECOUNTS and SAMPLESPERPIXEL tags. The reader should then check if the data block with this size (starting at the offset from the entry STRIPOFFSETS of the first channel) reaches behind the end of the file. If so, the calculated size should be reduced. Then the data can be read and be decompressed.
  This method works always since there is always one strip per image plane and channel stored in the file, the data for all channels of an image plane are always stored in adjacent regions of the file and the size of the uncompressed data is always larger than the size of the compressed data. Note that the latter is true for the block with all channels only. The “compressed” data of a single channel can be larger than the uncompressed data of that channel.

  With this method the reader will read more data than necessary. A second method is to determine the size of the compressed data by comparing the STRIPOFFSETS values of the current strip with the STRIPOFFSETS values of other strips and the size of the file.

- LSM files with 2 channels can have a deviation form the TIFF specification for the BITSPERSAMPLE tag. The writer used a separate region for the two 16-bit values, although they fit into the tag value element of the tag. Newer versions of the writer store three BITSPERSAMPLE values instead of two to overcome this problem. The reader should check the special case of 2 channels and should treat the tag value structure element as an offset, when more than 1 channel is stored in the image.

The whole file is stored in little endian byte order (Intel). For computers with other byte order (like Apple Macintosh) the byte order in all values has to be swapped.

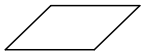
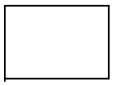
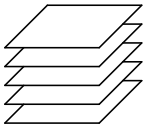
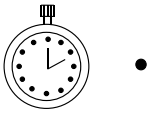
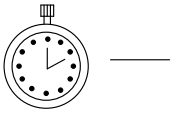
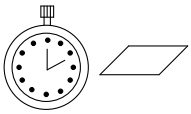
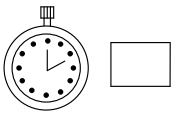
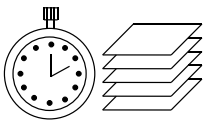
2. Image Data

The file starts with the TIFF-header

Header		
Type	Description	Value
UINT16	byte order	0x4949 (only intel order is supported)
UINT16	Identifier	42 (0x2a)
UINT32	Offset to the first image directory	

Table 1: TIFF Header

A LSM 5/7 file can contain more than one image directory. The last entry in the header contains the file offset to the first image directory. Table 1 contains the information how many image directories are present depending on the scan mode and which pixel data are handled by the directories. Every image directory handles all channels in every case.

	Scan mode "**Line**" - There is one image directory for pixel data and one thumbnail directory.
	Scan mode "**Plane**" - There is one image directory for pixel data and one thumbnail directory.
	Scan mode "**Z-Scan**" - There is one image directory for pixel data and one thumbnail directory. - One x-z plane is saved for each directory.
	Scan mode "**Stack**" - There are „z" image directories for pixel data and „z" thumbnail directories. - One x-y plane is saved for each directory.
	Scan mode "**Time Series Point**" (Introduced in release 3.0) In "Point" mode the intensities are stored in lines of 500 values. - There is one image directory for pixel data and one thumbnail directory. - One plane with 500 values in length is saved for each directory.
	Scan mode "**Time Series Line**" (Introduced in release 2.0) - There is one image directory for pixel data and one thumbnail directory. - One x-t plane is saved for each directory.
	Scan mode "**Time Series Plane**" - There are „t" image directories for pixel data and „t" thumbnail directories. - One x-y plane is saved for each directory.
	Scan mode "**Time Series z-Scan**" (Introduced in release 2.0) - There are „t" image directories for pixel data and „t" thumbnail directories. - One x-z plane is saved for each directory.
	Scan mode "**Time Series Stack**" (Introduced in release 2.3) - There are „z" * „t" image directories for pixel data and „z" * „t" thumbnail directories. - One x-y plane is saved for each directory. The first stack is stored in the first "z" directories. The second Stack is stored in directories z+1 to 2*z...

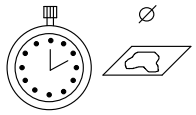
	Scan mode "**Time Series Mean-of-ROIs**" (Introduced in release 2.0) - 32-bit floating point values are saved - There is one image directory for pixel data and one thumbnail directory. - The saved plane has a width of the number of scanned ROIs and a height of the number of the time indices (ROI-t plane).

Table 2: Scan modes

For a spectral scan with the LSM 510 META detector or LSM 710 QUASAR detector the data for the different wavelengths are stored in different channels.

Release 5.5 (ZEN 2009) and later supports the acquisition at different positions and acquisition of several tiles. The two new dimensions P (position) and M (mosaic) have been introduced. The file contains the image directories for each position and tile. The order of the dimensions handled by the image directories is always X-Y-Z-T-C-P-M.

An image directory contains a 16-bit value with the number of directory entries followed by the directory entries with a length of 12 bytes each. The last field in the directory is an offset to the next image directory.

Type	Description	Value
UINT16	Number of tags	
12 Byte	Directory entry 1	
12 Byte	Directory entry 2	
…	…	
12 Byte	Directory entry n	
UINT32	Offset to the next image directory	„0“ in case of the last directory

Table 3: Image directory

A directory entry contains
- a tag describing the type of the entry
- the data type
- the number of values and
- a file offset to the start of the values.

Type	Description	Value	
UINT16	Tag	TIF_NEWSUBFILETYPE	(254)
		TIF_IMAGEWIDTH	(256)
		TIF_IMAGELENGTH	(257)
		TIF_BITSPERSAMPLE	(258)
		TIF_COMPRESSION	(259)
		TIF_PHOTOMETRICINTERPRETATION	(262)
		TIF_STRIPOFFSETS	(273)
		TIF_SAMPLESPERPIXEL	(277)
		TIF_STRIPBYTECOUNTS	(279)
		TIF_PLANARCONFIGURATION	(284)
		TIF_PREDICTOR	(317)
		TIF_COLORMAP	(320)
		TIF_CZ_LSMINFO	(34412)
UINT16	Type	TIF_BYTE	(0x0001)
		TIF_ASCII	(0x0002)
		TIF_SHORT	(0x0003)
		TIF_LONG	(0x0004)
		TIF_RATIONAL	(0x0005)
UINT32	Length	Number of values in *Value*	
UINT32	Value	File offset to the start of the values. If the all values require not more than 4 bytes, the values itself rather than an offset are stored in the location *Value*. The latter fact is important for the reader program. Depending on the type of the data sometimes the values fit into the 4 bytes and sometimes not.	

Table 4: Directory entry

Table 5 contains the list with all entries used in LSM 5/7 files.

TIF_NEWSUBFILETYPE	Type: Length: Value:	TIF_LONG 1 0 - for images 1 - for thumbnails
TIF_IMAGEWIDTH	Type: Length: Value:	TIF_LONG 1 Number of pixels in x-direction. Exceptions: Number of ROIs for scan mode "Time Series Mean-of-ROIs" and a value of 500 pixels for the scan mode "Point".
TIF_IMAGELENGTH	Type: Length: Value:	TIF_LONG 1 Depends on the scan mode: „**Plane**", „**Stack**", „**Time Series Plane**" and „**Time Series Stack**": Number of image pixels in y-direction "**z-Scan**" and "**Time Series z-Scan**": Number of image pixels in z-direction "**Time Series Line**" and "**Time Series Mean-of-ROIs**" Number of image pixels in t-direction " **Line**" and " **Point**": always 1.
TIF_BITSPERSAMPLE	Type: Length: Value:	TIF_SHORT Number of channels (1 or 3 for thumbnails). Number of bits per pixel for the corresponding channel 8 or 16 (for thumbnails: 8). The value is 32 for scan mode "Time Series Mean-of-ROIs". The value is also 32 for topography height map images.
TIF_COMPRESSION	Type: Length: Value:	TIF_SHORT 1 1 - no compression 5 - LZW-compression
TIF_PHOTOMETRIC-INTERPRETATION	Type: Length: Value:	TIF_SHORT 1 TIF_RGB24 (= 2) in case of more than one channel, TIF_PALETTE (=3) in case of one channel with color map and TIF_BLACKISZERO (=1) in case of one channel and without color map.
TIF_STRIPOFFSETS	Type: Length: Value:	TIF_LONG Number of channels („1" or „3" for thumbnails). Table with file offsets to the pixel data. For the LSM 5/7 format there is only one offset per channel.
TIF_SAMPLESPERPIXEL	Type: Length: Value:	TIF_SHORT 1 Number of channels („1" or „3" for thumbnails).

TIF_STRIPBYTECOUNTS	Type: TIF_LONG Length: Number of channels. Value: Table with the number of bytes in the pixel data strips. For the LSM 5/7 files: IMAGEWIDTH x IMAGELENGTH for 8 bit channel or thumbnail, IMAGEWIDTH x IMAGELENGTH x 2 for 12 or 16 bit channel. IMAGEWIDTH x IMAGELENGTH x 4 for 32 bit floating-point channel.
TIF_COLORMAP	Type: TIF_SHORT Length: 768 Value: Color palette with 256 entries for red followed by 256 entries for green followed by 256 entries for blue. This entry is only present if *TIF_HOTOMETRICINTERPRETATION* is *TIF_PALETTE*. The 8 bit color entries of the palette are stored in the upper 8 bits of each 16-bit word.
TIF_PLANARCON-FIGURATION	Type: TIF_SHORT Length: 1 Value: 2 (The channels are stored separately).
TIF_PREDICTOR	Type: TIF_SHORT Length: 1 Value: 2 (horizontal subtraction of neighbor pixel intensities prior the compression). This entry is only present for compressed pixel data *(TIF_COMPRESSION* == 5). The intensity of the first pixel in each image line is unmodified. For all following pixels in the image line the intensity difference to the pervious pixel was calculated. The resulting data are then compressed.
TIF_CZ_LSMINFO	Type: TIF_BYTE Length: Value: Structure with LSM-specific data (in the first image directory only).

Table 5: Tags used by the LSM 5/7

To access the pixel data the reader program has to use the array *TIF_STRIPOFFSETS* with file offsets to the image planes. For the LSM 5/7 format the image plane for one channel is stored in one continuous block and there is one file offset for each channel in the *TIF_STRIPOFFSETS* array (only one strip per image plane channel). Note that the reader in the LSM 5/7 program will not accept more than one strip per image plane channel.

3. CZ-Private Tag

The entry *TIF_CZ_LSMINFO* in the first image directory contains a file offset to a structure with detailed information of the image generation and the states of several editors. Basic information is stored directly in the structure. For additional information there are file offsets to further structures. See table 5 for a description of the structure.

UINT32 u32MagicNumber	0x00300494C (release 1.3) or 0x00400494C (release 1.5 to 6.0).
SINT32 s32StructureSize	Number of bytes in the structure.
SINT32 s32DimensionX	Number of intensity values in x-direction.
SINT32 s32DimensionY	Number of intensity values in y-direction.
SINT32 s32DimensionZ	Number of intensity values in z-direction or in case of scan mode “Time Series Mean-of-ROIs” the Number of ROIs.
SINT32 s32DimensionChannels	Number of channels.
SINT32 s32DimensionTime	Number of intensity values in time-direction.
SINT32 s32DataType	Format of the intensity values: 1 for 8-bit unsigned integer, 2 for 12-bit unsigned integer and 5 for 32-bit float (for “Time Series Mean-of-ROIs” or topography height map images) or 0 in case of different data types for different channels. In the latter case the field *32OffsetChannelDataTypes* contains further information.
SINT32 s32ThumbnailX	Width in pixels of a thumbnail.
SINT32 s32ThumbnailY	Height in pixels of a thumbnail.
FLOAT64 f64VoxelSizeX	Distance of the pixels in x-direction in meter.
FLOAT64 f64VoxelSizeY	Distance of the pixels in y-direction in meter.
FLOAT64 f64VoxelSizeZ	Distance of the pixels in z-direction in meter.
FLOAT64 f64OriginX	The x-offset of the center of the image in meter. relative to the optical axis. For LSM images the x-direction is the direction of the x-scanner. For cameras it is the CCD horizontal direction (depending on how camera is aligned/mounted) In releases prior 4.0 the entry was not used and the value 0 was written instead.
FLOAT64 f64OriginY	The y-offset of the center of the image in meter. relative to the optical axis. For LSM images the y-direction is the direction of the y-scanner. For cameras it is the CCD vertical direction (depending on how camera is aligned/mounted). In releases prior 4.0 the entry was not used and the value 0 was written instead.
FLOAT64 f64OriginZ	Not used.
UINT16 u16ScanType	Scan type 0 - normal x-y-z-scan 1 - z-Scan (x-z-plane) 2 - line scan 3 - time series x-y 4 - time series x-z (release 2.0 or later) 5 - time series “Mean of ROIs” (release 2.0 or later) 6 - time series x-y-z (release 2.3 or later) 7 - spline scan (release 2.5 or later) 8 - spline plane x-z (release 2.5 or later) 9 - time series spline plane x-z (release 2.5 or later) 10 - point mode (release 3.0 or later)

UINT16	u16SpectralScan	Spectral scan flag 0 – no spectral scan. 1 – image has been acquired in spectral scan mode with a LSM 510 META or LSM 710 QUASAR detector (release 3.0 or later).
UINT32	u32DataType	Data type 0 - Original scan data 1 - Calculated data 2 - 3D reconstruction 3 - Topography height map
UINT32	u32OffsetVectorOverlay	File offset to the description of the vector overlay (can be 0, if not present).
UINT32	u32OffsetInputLut	File offset to the channel input LUT with brightness and contrast properties (can be 0, if not present).
UINT32	u32OffsetOutputLut	File offset to the color palette (can be 0, if not present).
UINT32	u32OffsetChannelColors	File offset to the list of channel colors and channel names (can be 0, if not present).
FLOAT64	f64TimeIntervall	Time interval for time series in “s” (can be 0, if it is not a time series or if there is more detailed information in *u32OffsetTimeStamps*).
UINT32	u32OffsetChannelDataTypes	File offset to an array with *UINT32*-values with the format of the intensity values for the respective channels (can be 0, if not present). The contents of the array elements are 1 - for 8-bit unsigned integer, 2 - for 12-bit unsigned integer and 5 - for 32-bit float (for “Time Series Mean-of-ROIs”).
UINT32	u32OffsetScanInformation	File offset to a structure with information of the device settings used to scan the image (can be 0, if not present).
UINT32	u32OffsetKsData	File offset to “Zeiss Vision KS-3D” specific data (can be 0, if not present).
UINT32	u32OffsetTimeStamps	File offset to a structure containing the time stamps for the time indexes (can be 0, if it is not a time series).
UINT32	u32OffsetEventList	File offset to a structure containing the experimental notations recorded during a time series (can be 0, if not present).
UINT32	u32OffsetRoi	File offset to a structure containing a list of the ROIs used during the scan operation (can be 0, if not present).
UINT32	u32OffsetBleachRoi	File offset to a structure containing a description of the bleach region used during the scan operation (can be 0, if not present).
UINT32	u32OffsetNextRecording	For “Time Series Mean-of-ROIs” and for „Line scans“ it is possible that a second image is stored in the file (can be 0, if not present). For “Time Series Mean-of-ROIs” it is an image with the ROIs. For „Line scans“ it is the image with the selected line. In these cases *u32OffsetNextRecording* contains a file offset to a second file header. This TIFF-header and all sub-structures are built exactly the same way as a simple LSM 5/7 file. All offsets without exception are given there relative to the start of the second TIFF-header.
FLOAT64	f64DisplayAspectX	Zoom factor for the image display in x-direction (0.0 for release 2.3 and earlier).
FLOAT64	f64DisplayAspectY	Zoom factor for the image display in y-direction (0.0 for release 2.3 and earlier).

FLOAT64 f64DisplayAspectZ	Zoom factor for the image display in z-direction (0.0 for release 2.3 and earlier).
FLOAT64 f64DisplayAspectTime	Zoom factor for the image display in time-direction (0.0 for release 2.3 and earlier).
UINT32 u32OffsetMeanOfRoisOverlay	File offset to the description of the vector overlay with the ROIs used during a scan in “Mean of ROIs” mode (can be 0, if not present).
UINT32 u32OffsetTopoIsolineOverlay	File offset to the description of the vector overlay for the topography–iso–lines and height display with the profile selection line (can be 0, if not present).
UINT32 u32OffsetTopoProfileOverlay	File offset to the description of the vector overlay for the topography–profile display (can be 0, if not present).
UINT32 u32OffsetLinescanOverlay	File offset to the description of the vector overlay for the line scan line selection with the selected line or Bezier curve (can be 0, if not present).
UINT32 u32ToolbarFlags	Bit-field for disabled toolbar buttons: bit 0 - “Corp” button bit 1 - “Reuse” button. If the bit is set the corresponding button is disabled.
UINT32 u32OffsetChannelWavelength	Offset to memory block with the wavelength range used during acquisition for the individual channels (new for release 3.0; can be 0, if not present).
UINT32 u32OffsetChannelFactors	Offset to memory block with scaling factor, offset and unit for each image channel. The data are currently used by images with ion concentration data. The display value is calculated by DisplayValue = Factor * PixelIntensity / MaxPixelIntensity + Offset. where "MaxPixelIntensity" is the maximum possible pixel intensity for the data type (4095 or 255). The parameters are stored in an array of structures LSMCHANNELFACTORS (24 bytes per channel) with the members: FLOAT64 f64Factor FLOAT64 f64Offset UINT32 u32Unit UINT32 u32Reserved [3] u32Unit can be: *eUnitNone* – unknown or *eUnitConcentration* - Ion concentration in mol.
FLOAT64 f64ObjectiveSphereCorrection	The inverse radius of the spherical error of the objective that was used during acquisition. This is the radius of the sphere that can be fitted on the topography reconstruction of an absolutely plane object that has been recorded with the objective.
UINT32 u32OffsetUnmixParameters	File offset to the parameters for linear unmixing that have been used to generate the image data from scan data of the spectral detector (new for release 3.2; can be 0, if not present).
UINT32 u32OffsetAcquisitionParameters	File offset to a block with acquisition parameters for support of the re-use function of the LSM 5/7 program

		(new for release 3.5; can be 0, if not present).
UINT32	u32OffsetCharacteristics	File offset to a block with user specified properties (new for release 3.5; can be 0, if not present).
UINT32	u32OffsetPalette	File offset to a block with detailed color palette properties (new for release 3.5; can be 0, if not present).
FLOAT64	f64TimeDifferenceX	The time difference for the acquisition of adjacent pixels in x-direction in seconds. The property is used by RICS analysis (new for release 5.0; can be 0, if not present).
FLOAT64	f64TimeDifferenceY	The time difference for the acquisition of adjacent pixels in y-direction in seconds. The property is used by RICS analysis (new for release 5.0; can be 0, if not present).
FLOAT64	f64TimeDifferenceZ	The time difference for the acquisition of adjacent pixels in z-direction in seconds. The property is used by RICS analysis (new for release 5.0; can be 0, if not present).
UINT32	u32InternalUse1	Reserved for internal use. Writer should set this field to 0.
SINT32	s32DimensionP	Number of intensity values in position-direction. (new for release 5.5; can be 0, if not present).
SINT32	s32DimensionM	Number of intensity values in tile (mosaic)-direction. (new for release 5.5; can be 0, if not present).
SINT32	s32DimensionsReserved[16]	16 reserved 32-bit words, must be 0.
UINT32	u32OffsetTilePositions	File offset to a block with the positions of the tiles in (new for release 5.5; can be 0, if not present).
UINT32	u32Reserved [9]	9 reserved 32-bit words, must be 0.
UINT32	u32OffsetPositions	File offset to a block with the positions of the acquisition regions (new in release 6.2, can be 0, if not present, release 6.2 is NOT released yet! beta and special built only).
UINT32	u32Reserved [21]	21 reserved 32-bit words, must be 0.

Table 6 Structure with LSM-specific data

4. Channel Names and Colors

The structure with information about the channel names and colors is accessible via the *u32OffsetChannelColors* entry of the CZ-Private tag.

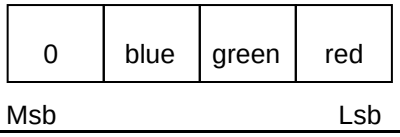
SINT32	s32BlockSize	Size of the structure in bytes including the name strings and colors.
SINT32	s32NumberColors	Number of colors in the color array; should be the same as the number of channels.
SINT32	s32NumberNames	Number of character strings for the channel names; should be the same as the number of channels.
SINT32	s32ColorsOffset	Offset to the “UINT32” array of channel colors relative to the start of the structure. Each array entry contains a color with intensity values in the range 0..255 for the three color components 
SINT32	s32NamesOffset	Offset to the list of channel names relative to the start of the structure. The list of channel names is a series of “\0”-terminated ANSI character strings.
SINT32	s32Mono	If unequal zero the “Mono” button in the LSM-image window was pressed
SINT32	s32Reserved[4]	Four 32-bit words reserved for use in the future. The values are set to “0”.
... Colors ...		
... Channels names ...		

Table 7: Structure with channel name Contents s and colors

5. Drawing Elements for Vector Overlay, ROIs and Bleach ROIs

The LSM 5/7 program uses the same structures for vector overlays, selection lines the scan ROIs and bleach ROIs. The main difference is that for line selection displays only lines and for the ROIs only closed drawing elements (rectangles, circles ellipses, polygons and closed Bezier curves) are allowed.

The “CZ-Private tag” contains file offsets to the start of the structures:

u32OffsetVectorOverlay - for the vector overlay,
u32OffsetRoi - for the scan ROIs,
u32OffsetBleachRoi - for the bleach ROIs.
u32OffsetMeanOfRoisOverlay - for the ROIs used during a scan in “Mean of ROIs” mode,
u32OffsetTopoIsolineOverlay - for the topography display with the profile selection line,
u32OffsetTopoProfileOverlay - for the overlay in topography profile display and
u32OffsetLinescanOverlay - for the overlay with the line scan selection line or Bezier curve.

The file region starts with a header. The header is followed by the list of the drawing elements.

SINT32	s32NumberDrawingElements	Number of drawing elements in the list.
SINT32	s32Size	Size of the whole description block including drawing elements and header in bytes.
SINT32	s32LineWidth	Line width, most recently used in the editor in pixels. The value is used for new generated drawing elements in the LSM program.
SINT32	s32Measure	Unequal „0“ if the „measure“-button was checked in the editor, else „0“.
SINT32	s32Reserved1	Must be „0“.
SINT32	s32Reserved2	Must be „0“.
SINT32	s32Color	Color most recently used in the editor with intensity values in the range 0..255 for the three color components 0 | blue | green | red Msb Lsb
SINT32	s32Valid	„1“ if the „overlay“ was enabled in the editor and “0” if not. This value should be set to a value equal to „1“ for ROIs because there is no „disable“ button in the ROI editors and the ROIs would be invisible.
SINT32	s32KnotWidth	Size of the knots for modification of the drawing elements. The value is used for new generated drawing elements in the LSM program. A knot is „2 * s32KnotWidth“ pixels wide and high.
SINT32	s32CatchArea	Size of the mouse catch area for activation of a drawing element with a mouse click. The value is used for new generated drawing elements in the LSM program.
SINT32	s32FontHeight	These fields are used by the „Windows Font Mapper“
SINT32	s32FontWidth	to select the font for new generated „text“ drawing
SINT32	s32FontEscapement	elements. The meaning is the same as for the
SINT32	s32FontOrientation	corresponding fields described in „Table 8“.
SINT32	s32FontWeight	
SINT32	s32FontItalic	
SINT32	s32FontUnderline	
SINT32	s32FontStrikeOut	

SINT32	s32FontCharSet	
SINT32	s32FontOutPrecision	
SINT32	s32FontClipPrecision	
SINT32	s32FontQuality	
SINT32	s32FontPitchAndFamily	
UINT16	u16FontFaceName [32]	
UINT8	u8MeasureFlags[10]	Array with flags indicating what kind of measure values have to be displayed for the different drawing element types. These flags are used for new created drawing elements by the overlay editor. The array index is the type of the drawing element: 0 - closed polyline 1 - open polyline 2 - closed Bezier curve 3 - open Bezier curve 4 - arrow with closed tip 5 - arrow with open tip 6 – ellipse 7 – circle 8 – rectangle 9 – line The flags are: 0x02 – circumference 0x04 – area 0x08 – radius 0x10 - angle 0x20 – distance x 0x40 – distance y If the array element is “0” the default flags for the drawing element type are used.
SINT32	s32Reserved [7]	7 reserved 32-bit words; Must be 0.
Drawing element 1		
Drawing element 2		
... ...		
Drawing element	s32NumberDrawingElements	

Table 8: Structure for drawing element lists (header and drawing elements)

The description of a particular drawing element contains two parts. The first part is a fixed sized structure with equal fields for all drawing elements. The second part is drawing element type specific.

The fixed part is different in release 1.3 and the later releases. The main difference is that in release 1.3 all coordinates were given in integer coordinates and in later versions in floating point coordinates. It is possible to distinguish the kind of the structure by the *s32Type* field, which is the first field in all releases.

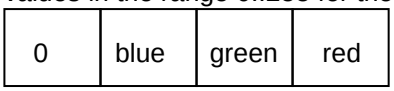
Note that releases prior 2.0 are not able to handle “ROIs” and “Bleach ROIs”. If the reader program is interested in ROIs only, there is no need to regard the structures of version 1.3.

All horizontal coordinates are given in horizontal image memory coordinates.

For all scan modes except “z-Scan“ and „time series z-scan“ the vertical coordinates are given in vertical scan memory coordinates. For „z-scan“ and „time series z-scan“, a vertical coordinate in scan memory coordinates must be calculated by

ScanMemoryVertCoordinate = DrawingElementVertCoordinate * f64VoxelSizeX / f64VoxelSizeZ

SINT32 s32Type	Type of the drawing element in release 1.5 or later: 13 – Text (DRAWING_ELEMENT_FLOAT_TEXT) 14 – Line (DRAWING_ELEMENT_FLOAT_LINE) 15 – Horizontal or vertical scale bar with displayed length (DRAWING_ELEMENT_FLOAT_SCALE_BAR). 16 – Line with two line arrow tip (DRAWING_ELEMENT_FLOAT_OPEN_ARROW) 17 – Line with three line arrow tip (DRAWING_ELEMENT_FLOAT_CLOSED_ARROW) 18 – Rectangle (DRAWING_ELEMENT_FLOAT_RECTANGLE) 19 – Ellipse (DRAWING_ELEMENT_FLOAT_ELLIPSE) 20 – Closed polyline (DRAWING_ELEMENT_FLOAT_CLOSED_POLYLINE) 21 – Open polyline (DRAWING_ELEMENT_FLOAT_OPEN_POLYLINE) 22 – Closed Bezier spline curve (DRAWING_ELEMENT_FLOAT_CLOSED_BEZIER) 23 – Open Bezier spline curve (DRAWING_ELEMENT_FLOAT_OPEN_BEZIER) 24 – Circle (DRAWING_ELEMENT_FLOAT_CIRCLE) 25 – Rectangle with color palette (DRAWING_ELEMENT_FLOAT_PALETTE) 26 – Open polyline with arrow tip (DRAWING_ELEMENT_FLOAT_POLYLINE_ARROW) 27 – Open Bezier spline curve with arrow tip (DRAWING_ELEMENT_FLOAT_BEZIER_WITH_ARROW) 28 – Two connected lines for angle measurement (DRAWING_ELEMENT_FLOAT_ANGLE) 29 – Circle defined by three points on the perimeter (DRAWING_ELEMENT_FLOAT_CIRCLE_3POINT) Table 10 contains the drawing element type identifiers for version 1.3.
SINT32 s32Size	Size of the block for the description of the current drawing element in bytes. This value should be used to find out the start of the next drawing element block.
SINT32 s32LineWidth	Line width used to draw the element in pixels.
SINT32 s32Measure	The value 0 indicates that no measured characteristics are drawn for the drawing element. The value 1 indicates that the default set of measured characteristics is displayed. If the value is not 0 and not 1 the value contains flags for enabled types of measure values. The flags are: 0x00000002 – circumference 0x00000004 – area 0x00000008 – radius 0x00000010 - angle 0x00000020 – distance x

		0x00000040 – distance y Flags, which are not applicable for a particular type of drawing elements, have no effect.
FLOAT64	f64AdditTextStartPointX	Horizontal start of additional text in image memory coordinates. This can be the text with „measured characteristics“ for the „overlay“ or the ROI number for ROIs.
FLOAT64	f64AdditTextStartPointY	Vertical start of additional text in image memory coordinates. This can be the text with „measured characteristics“ for the „overlay“ or the ROI number for ROIs.
SINT32	s32Color	Specifies the color used to draw the element with intensity values in the range 0..255 for the three color components  Msb Lsb
SINT32	s32Valid	If unequal „zero“ the drawing element was completely edited. This value should be unequal „zero“.
SINT32	s32KnotWidth	Size of the knots used to modify the drawing element. A knot is „2 * s32KnotWidth“ pixels wide and high.
SINT32	s32CatchArea	Size of the mouse catch area for activation of the drawing element with a mouse click.
SINT32	s32FontHeight	Height of the font for text drawing. The value is given in logical units which, for the usual graphics cards, mean that the height in pixels is: „1.6 * s32FontHeight“. All font properties are used only for elements of type „DRAWING_ELEMENT_FLOAT_TEXT“ by the windows font mapper to select the font for text drawing.
SINT32	s32FontWidth	Average width of characters in the font in logical units. If this value is zero a default width is used for the font.
SINT32	s32FontEscapement	Angle of each line of text written in the font in tenths of degrees. The default value is 0.
SINT32	s32FontOrientation	Angle of each character's base line in tenths of degrees. The default value is 0.
SINT32	s32FontWeight	Weight of the font in the range 0 through 1000 (for example, 400 is normal and 700 is bold). If the value is zero, a default weight is used.
SINT32	s32FontItalic	Italic font if set to a value unequal „0“. The default value is 0.
SINT32	s32FontUnderline	Underlined font if set to a value unequal „0“. The default value is 0.
SINT32	s32FontStrikeOut	Strikeout font if set to a value unequal „0“. The default value is 0.
SINT32	s32FontCharSet	Type of font required to draw the text correctly. The value should be set to 0.
SINT32	s32FontOutPrecision	The output precision. The output precision defines how closely the output must match the requested font's height, width, character orientation, escapement, and pitch. The value should be 0.
SINT32	s32FontClipPrecision	The clipping precision. The clipping precision defines how to clip characters that are partially outside the clipping region. The value should be 0.
SINT32	s32FontQuality	Specifies how carefully the system must attempt to match the font attributes to those of an actual physical font. The value should be 0.
SINT32	s32FontPitchAndFamily	Specifies the pitch and family of the font. The two low-order bits specify the pitch of the font and can be one of the following values:

	0 - system default pitch 1 - fixed pitch 2 - variable pitch. Bits 4 through 7 of the member specify the font family and can be one of the following values: 0 - don't care (0<<4) 1 - variable stroke width, serifed (1<<4) 2 - variable stroke width, sans-serifed (2<<4) 3 - constant stroke width, serifed or sans-serifed (3<<4) 4 - cursive (4<<4) 5 - old English (5<<4) The proper value can be obtained by using the boolean OR operator to join one pitch constant with one family constant.
UINT16 u16FontFaceName [32]	Zero terminated UNICODE character string with the name of the font to use (UNICODE characters have two bytes per character. The values 0 to 127 have the same meaning as the corresponding values in the ANSI character set).
UINT16 s32Disabled	If equal „zero" the drawing element is enabled (visible). Drawing elements in the overlay are always enabled. ROIs can be disabled.
SINT32 s32NotMoveable	If this value is not zero the drawing element can not be moved with the editor.
SINT32 s32Reserved [8]	8 reserved 32 bit words. Must be "0".

Table 8: Fixed part of the drawing element structure for versions higher than 1.3

The variable part is follows immediately after the fixed structure part.

DRAWING_ELEMENT_FLOAT_TEXT :

FLOAT64 f64PointOriginX	Horizontal coordinate of the start of the text (left border of the text bounding rectangle)
FLOAT64 f64PointOriginX	Vertical coordinate of the start of the text (upper border of the text bounding rectangle)
UINT16 u16Text [ ]	Zero terminated UNICODE character string with the text to draw. (UNICODE characters have two bytes per character. The values 0 to 127 have the same meaning as the corresponding values in the ANSI character set).

DRAWING_ELEMENT_FLOAT_LINE :
DRAWING_ELEMENT_FLOAT_SCALE_BAR :
DRAWING_ELEMENT_FLOAT_OPEN_ARROW :
DRAWING_ELEMENT_FLOAT_CLOSED_ARROW :

UINT32 u32NumberKnots	The number of knots. This value must be 2.
FLOAT64 f64Knot1X	Horizontal coordinate of the start point.
FLOAT64 f64Knot1Y	Vertical coordinate of the start point.
FLOAT64 f64Knot2X	Horizontal coordinate of the end point.
FLOAT64 f64Knot2Y	Vertical coordinate of the end point.

DRAWING_ELEMENT_FLOAT_RECTANGLE :
DRAWING_ELEMENT_FLOAT_PALETTE :

UINT32 u32NumberKnots	The number of knots. This value must be 2.
FLOAT64 f64Knot1X	Horizontal coordinate of the first of the two diagonal arranged corners.
FLOAT64 f64Knot1Y	Vertical coordinate of the first of the two diagonal arranged corners.
FLOAT64 f64Knot2X	Horizontal coordinate of the second of the two diagonal arranged corners.
FLOAT64 f64Knot2Y	Vertical coordinate of the second of the two diagonal arranged corners.

DRAWING_ELEMENT_FLOAT_ELLIPSE :

UINT32 u32NumberKnots	The number of knots. This value must be 4.
FLOAT64 f64Knot1X	Knot1 and Knot3 are the two cut points of the ellipse with
FLOAT64 f64Knot1Y	one of the main axes and Knot2 and Knot4 the cut points
FLOAT64 f64Knot2X	with the other main axis.
FLOAT64 f64Knot2Y	f64Knot1X .. f64Knot4X are the horizontal coordinates
FLOAT64 f64Knot3X	and
FLOAT64 f64Knot3Y	f64Knot1Y... f64Knot4Y are the vertical coordinates.
FLOAT64 f64Knot4X	
FLOAT64 f64Knot4Y	

DRAWING_ELEMENT_FLOAT_CIRCLE :

UINT32 u32NumberKnots	The number of knots. This value must be 2.
FLOAT64 f64Knot1X	Horizontal coordinate of the center point.
FLOAT64 f64Knot1Y	Vertical coordinate of the center point.
FLOAT64 f64Knot2X	Horizontal coordinate of one curve point.
FLOAT64 f64Knot2Y	Vertical coordinate of the curve point where the horizontal coordinate is f64Knot2X.

DRAWING_ELEMENT_FLOAT_CIRCLE_3POINT :

UINT32 u32NumberKnots	The number of knots. Must be 3.
FLOAT64 f64Knot1X	coordinates of the three points on the perimeter.
FLOAT64 f64Knot1Y	f64Knot1X .. f64Knot3X are the horizontal coordinates
FLOAT64 f64Knot2X	and
FLOAT64 f64Knot2Y	f64Knot1Y... f64Knot3Y are the vertical coordinates.
FLOAT64 f64Knot3X	
FLOAT64 f64Knot3Y	

DRAWING_ELEMENT_FLOAT_ANGLE :

UINT32 u32NumberKnots	The number of knots. Must be 3.
FLOAT64 f64Knot1X	coordinates of the three knots.
FLOAT64 f64Knot1Y	f64Knot1X .. f64Knot3X are the horizontal coordinates
FLOAT64 f64Knot2X	and
FLOAT64 f64Knot2Y	f64Knot1Y... f64Knot3Y are the vertical coordinates.
FLOAT64 f64Knot3X	
FLOAT64 f64Knot3Y	

DRAWING_ELEMENT_FLOAT_CLOSED_POLYLINE
DRAWING_ELEMENT_FLOAT_OPEN_POLYLINE
DRAWING_ELEMENT_FLOAT_POLYLINE_ARROW :

UINT32 u32NumberKnots	The number of vertices.
FLOAT64 f64Knot1X	Knot1 ... KnotN are the vertices of the polyline.
FLOAT64 f64Knot1Y	f64Knot1X .. f64KnotNX are the horizontal coordinates
FLOAT64 f64Knot2X	and
FLOAT64 f64Knot2Y	f64Knot1Y... f64KnotNY are the vertical coordinates.
...	
...	
FLOAT64 f64KnotNX	
FLOAT64 f64KnotNY	

DRAWING_ELEMENT_FLOAT_CLOSED_BEZIER
DRAWING_ELEMENT_FLOAT_OPEN_BEZIER
DRAWING_ELEMENT_FLOAT_BEZIER_WITH_ARROW :

UINT32 u32NumberKnots	The number of knots.
FLOAT64 f64Knot1X	Knot1 ... KnotN are knots arranged by the user.
FLOAT64 f64Knot1Y	f64Knot1X .. f64KnotNX are the horizontal coordinates
FLOAT64 f64Knot2X	and
FLOAT64 f64Knot2Y	f64Knot1Y... f64KnotNY are the vertical coordinates.
...	These knots are the endpoints of cubic Bezier curve
...	segments. The two control points for the Bezier curve
FLOAT64 f64KnotNX	segments are calculated with the assumption of
FLOAT64 f64KnotNY	equidistant parameterization of the curve.

In release 1.3 the fixed header had no “s32Enabled” field and all floating-point coordinates in the structures of release 1.5 or later have been integer coordinates in release 1.3.

SINT32	s32Type
SINT32	s32Size
SINT32	s32LineWidth
SINT32	s32Measure
SINT32	s32AdditTextStartPointX
SINT32	s32AdditTextStartPointY
SINT32	s32Color
SINT32	s32Valid
SINT32	s32KnotWidth
SINT32	s32CatchArea
SINT32	s32FontHeight
SINT32	s32FontWidth
SINT32	s32FontEscapement
SINT32	s32FontOrientation
SINT32	s32FontWeight
SINT32	s32FontItalic
SINT32	s32FontUnderline
SINT32	s32FontStrikeOut
SINT32	s32FontCharSet
SINT32	s32FontOutPrecision
SINT32	s32FontClipPrecision
SINT32	s32FontQuality
SINT32	s32FontPitchAndFamily
UINT16	u16FontFaceName [32]
SINT32	s32Reserved [10]

Table 9. Fixed part of the drawing element structure for release 1.3

Type value	Symbol	Description
1	DRAWING_ELEMENT_TEXT	Text
2	DRAWING_ELEMENT_LINE	Line
3	DRAWING_ELEMENT_RECTANGLE	Rectangle
4	DRAWING_ELEMENT_ELLIPSE	Ellipse
5	DRAWING_ELEMENT_CIRCLE	Circle
6	DRAWING_ELEMENT_CLOSED_POLYLINE	Closed polyline
7	DRAWING_ELEMENT_OPEN_POLYLINE	Open polyline
8	DRAWING_ELEMENT_CLOSED_BEZIER	Closed Bezier spline curve
9	DRAWING_ELEMENT_OPEN_BEZIER	Open Bezier spline curve
10	DRAWING_ELEMENT_CLOSED_ARROW	Line and a three line arrow
11	DRAWING_ELEMENT_OPEN_ARROW	Line and a two line arrow
12	DRAWING_ELEMENT_SCALE_BAR	Horizontal or vertical scale bar with displayed length

Table 10: Drawing element type identifiers for version 1.3

6. Time Stamp Information

Time stamp information is available in Release 2.0 or later and is present only for time series scan modes.

The field “u32OffsetTimeStamps” of the “CZ-Private tag” contains the file offsets to the start of the time stamp structure.

SINT32	s32Size	Size, in bytes, of the whole block used for time stamps.
SINT32	s32NumberTimeStamps	Number of time stamps in the list.
FLOAT64	f64TimeStamp1	Time stamps in seconds relative to the start time of
FLOAT64	f64TimeStamp2	the LSM electronic unit controller program.
...		
FLOAT64	f64TimeStampN	

Table 11 Structure for time stamp information

7. Event List

The event list is available in Release 2.0 or later for time series scan modes.

The field *u32OffsetEventList* of the “CZ-Private tag” contains the file offsets to the start of the time stamp structure.

SINT32	s32Size	Size of the whole block in bytes.
SINT32	s32NumberEvents	Number of recorded events in the list.
Event1		Event list entries
Event2		
...		
Eventn		

Table 12 Structure for the event list

UINT32	u32SizeEventListEntry	Size of the entry in bytes.
FLOAT64	f64Time	Time of the event in seconds relative to the start time of the LSM electronic module controller program.
UINT32	u32EventType	Can be one of the following values: EV_TYPE_MARKER (= 0) - Experimental annotation EV_TYPE_TIMER_CHANGE (= 1) - The time interval has changed EV_TYPE_BLEACH_START (= 2) - Start of a bleach operation EV_TYPE_BLEACH_STOP (= 3) - End of a bleach operation EV_TYPE_TRIGGER (= 4) - A trigger signal was detected on the user port of the electronic module.
UINT8	u8Description []	Zero terminated character string with the event description.

Table 13: Event list entry

8. Lookup Tables

The input lookup tables (with brightness, contrast, gamma and user defined settings) and the color palette (output lookup table) is treated the same way in the LSM 5/7 file format.

The fields *u32OffsetInputLut* and *u32OffsetOutputLut* of the "CZ-Private tag" contains the file offsets to the start of the input and output lookup table description structures.

UINT32	u32Size	Size of the whole description block including this header in bytes.
UINT32	u32NumberSubBlocks	Number of sub blocks without the List End Marker sub block "SUBBLOCK_LIST_END".
UINT32	u32NumberChannels	Number of channels handled in the description block. For the color palette it must be 3 and for the input lookup table it must be the number of data channels.
UINT32	u32LutType	Most recently selected type of the LUT LUT_NORMAL = 0 LUT generated by the brightness and contrast properties. LUT_ORIGINAL = 1 LUT generated by 12 bit to 12 bit LUT and the brightness and contrast properties. LUT_RAMP = 2 LUT generated by a ramp defined by start and end point, the brightness and the contrast properties. LUT_POLYLINE = 3 LUT generated by multiple lines, the brightness and the contrast properties. LUT_SPLINE = 4 LUT generated by a cubic spline curve, the brightness and the contrast properties. LUT_GAMMA = 5 LUT generated by an exponential function, the brightness and the contrast properties (Not used in case of color palette).
UINT32	u32Advanced	This flag is used for the input LUT and is unequal zero if the "More/Simple" button in the "Brightness and Contrast" dialog was pressed.
UINT32	u32CurrentChannel	Zero based number of the channel that was most recently selected for modifications in the LUT editor. A value of "-1" means all channels were selected.
UINT32	u32Reserved [9]	Reserved, must be 0

Table 14 Lookup table header

The list of sub blocks follows immediately after the header. The first entry in every sub block is a sub block type. The second entry specifies the size of the sub block, which can be used to find out the start of the next sub block. The list must end with *SUBBLOCK_LIST_END* The sub blocks are described in the tables below.

UINT32	u32Type	SUBBLOCK_GAMMA = 1
UINT32	u32Size	Size of the sub block in bytes.
FLOAT64 FLOAT64 ... FLOAT64	f64GammaChannel1 f64GammaChannel2 f64GammaChannelN	Exponent of the exponential function for the corresponding channels 1..N.. LUT[Intens] = MaxIntens *(pow(Intens/MaxIntens,Gamma) * 0.5/(1-Contrast)+Brightness)

UINT32	u32Type	SUBBLOCK_BRIGHTNESS = 2
UINT32	u32Size	Size of the sub block in bytes.
FLOAT64 FLOAT64 ... FLOAT64	f64BrightnessChannel1 f64BrightnessChannel2 f64BrightnessChannelN	Brightness values for the corresponding channels 1..N in the range 0..1. LUT[Intens] = f(Intens) *0.5 /(1-Contrast)+Brightness

UINT32	u32Type	SUBBLOCK_CONTRAST = 3
UINT32	u32Size	Size of the sub block in bytes.
FLOAT64 FLOAT64 ... FLOAT64	f64ContrastChannel1 f64ContrastChannel2 f64ContrastChannelN	Value representing the contrast for the corresponding channels 1..N in the range 0..1. LUT[Intens] = f(Intens) *0.5 /(1-Contrast)+Brightness

UINT32	u32Type	SUBBLOCK_RAMP = 4
UINT32	u32Size	Size of the sub block in bytes.
FLOAT64 FLOAT64 FLOAT64 FLOAT64 “ “ “ “ Channel2 ... “ “ “ “ ChannelN	f64RampStartXChannel1 f64RampStartYChannel1 f64RampEndXChannel1 f64RampEndYChannel1	Start and end points for the ramp for the corresponding channels. The range is 0..255 for both coordinates. LUT [Intens] = MaxIntens / 255 * StartY, if Intens < StartX LUT [Intens] = MaxIntens / 255 * EndY, if Intens > EndX LUT [Intens] = MaxIntens / 255 * ((EndY - StartY) * (Intens - StartX) / (EndX - StartX) + StartY)

UINT32	u32Type	SUBBLOCK_KNOTS = 5
UINT32	u32Size	Size of the sub block in bytes.
Knots for channel 1 FLOAT64 f64Knot1X FLOAT64 f64Knot1Y FLOAT64 f64Knot2X FLOAT64 f64Knot2Y .... FLOAT64 f64KnotNX FLOAT64 f64KnotNY Knots for channel 2 ... Knots for channel M		The vertices used for a polyline or the knots used for cubic spline curve shaped LUT. The range is 0..255 for both coordinates. The x-coordinate of the first knot must be 0. The x-coordinate of the last knot must be 255.

UINT32	u32Type	SUBBLOCK_PALETTE_12_TO_12 = 6
UINT32	u32Size	Size of the sub block in bytes.
Palette for channel 1 SINT16 s16Intensity [4096] Palette for channel 2 SINT16 s16Intensity [4096] ... Palette for channel N SINT16 s16Intensity [4096]		12 bit to 12 bit LUT for the corresponding channels 1..N. LUTres [Intens] = MaxIntens / 255 * (LUT [Intens] * 0.5 /(1-Contrast)+Brightness)

UINT32	u32Type	SUBBLOCK_LIST_END = 0

9. Acquisition Information

The field *u32OffsetScanInformation* of the “CZ-Private tag” contains the file offset to the start of a block with information of the device settings used during acquisition. Note that the image size, channel number and pixel distance of the image stored in the file can be different form the settings used during acquisition because the image could have been modified by offline operations. Information about the image size, channel number and pixel distance of the image contents stored in the file can be found in the fields of the “CZ-Private tag”.

The acquisition information is a list with entries of variable length. Each entry has the structure described in Table 15.

UINT32	u32Entry	A value that specifies which data are stored
UINT32	u32Type	A value that specifies the type of the data stored in the „Variable length data“ field. TYPE_SUBBLOCK - start or end of a sub block TYPE_LONG - 32 bit signed integer TYPE_RATIONAL - 64 bit floating point TYPE_ASCII - zero terminated string.
UINT32	u32Size	Size, in bytes, of the „Variable length data“ field.
Variable length data		

Table 15: Entries in the acquisition information list

The acquisition information contains hierarchical ordered information. To map this structure in a linear file a list of sub blocks is used. Every sub block starts with a list entry of the type TYPE_SUBBLOCK. The u32Entry field can be:

SUBBLOCK_RECORDING	= 0x010000000
SUBBLOCK_LASERS	= 0x030000000
SUBBLOCK_LASER	= 0x050000000
SUBBLOCK_TRACKS	= 0x020000000
SUBBLOCK_TRACK	= 0x040000000
SUBBLOCK_DETECTION_CHANNELS	= 0x060000000
SUBBLOCK_DETECTION_CHANNEL	= 0x070000000
SUBBLOCK_ILLUMINATION_CHANNELS	= 0x080000000
SUBBLOCK_ILLUMINATION_CHANNEL	= 0x090000000
SUBBLOCK_BEAM_SPLITTERS	= 0x0A0000000
SUBBLOCK_BEAM_SPLITTER	= 0x0B0000000
SUBBLOCK_DATA_CHANNELS	= 0x0C0000000
SUBBLOCK_DATA_CHANNEL	= 0x0D0000000
SUBBLOCK_TIMERS	= 0x011000000
SUBBLOCK_TIMER	= 0x012000000
SUBBLOCK_MARKERS	= 0x013000000
SUBBLOCK_MARKER	= 0x014000000

Every sub block ends with a list entry of the type TYPE_SUBBLOCK and an entry value of SUBBLOCK_END (=0x0FFFFFFFF).

Some of the entries were used in previous releases only. These entries are not described if they contained always the default value or if the value was not used at all, so that there is absolutely no effect if the entry is not present and the file is read by older LSM program versions.

An example for the sequence of sub blocks is:

```
SUBBLOCK_RECORDING
        Recording information ...
        SUBBLOCK_LASERS
                SUBBLOCK_LASER
                        Information Laser 1...
                SUBBLOCK_END
                SUBBLOCK_LASER
                        Information Laser 2...
                SUBBLOCK_END
                ...
        SUBBLOCK_END
        SUBBLOCK_TRACKS
                SUBBLOCK_TRACK
                        Information Track 1...
                        SUBBLOCK_DETECTION_CHANNELS
                                SUBBLOCK_DETECTION_CHANNEL
                                        „nformation Detection Channel 1...
                                SUBBLOCK_END
                                SUBBLOCK_DETECTION_CHANNEL
                                        Information Detection Channel 2...
                                SUBBLOCK_END
                                ...
                        SUBBLOCK_END
                        SUBBLOCK_ILLUMINATION_CHANNELS
                                SUBBLOCK_ILLUMINATION_CHANNEL
                                        Information Illumination Channel 1...
                                SUBBLOCK_END
                                SUBBLOCK_ILLUMINATION_CHANNEL
                                        Information Illumination Channel 2...
                                SUBBLOCK_END
                                ...
                        SUBBLOCK_END
                        SUBBLOCK_BEAM_SPLITTERS
                                SUBBLOCK_BEAM_SPLITTER
                                        Information Beam Splitter 1....
                                SUBBLOCK_END
                                SUBBLOCK_BEAM_SPLITTER
                                        Information Beam Splitter 2...
                                SUBBLOCK_END
                                ...
                        SUBBLOCK_END
                        SUBBLOCK_DATA_CHANNELS
                                SUBBLOCK_DATA_CHANNEL
                                        Information Data Channel 1...
                                SUBBLOCK_END
                                SUBBLOCK_DATA_CHANNEL
                                        Information Data Channel 2...
                                SUBBLOCK_END
                                ...
                        SUBBLOCK_END
                SUBBLOCK_END
                SUBBLOCK_TRACK
                        Information Track 2...
                        ....
                SUBBLOCK_END
        SUBBLOCK_END
        SUBBLOCK_TIMERS
                SUBBLOCK_TIMER
                        Information Timer 1...
                SUBBLOCK_END
                SUBBLOCK_TIMER
                        Information Timer 2...
                SUBBLOCK_END
                ...
        SUBBLOCK_END
        SUBBLOCK_MARKERS
                SUBBLOCK_MARKER
                        Information Marker 1...
                SUBBLOCK_END
                SUBBLOCK_MARKER
                        Information Marker 2...
                SUBBLOCK_END
                ...
        SUBBLOCK_END
SUBBLOCK_END
```

9.1. Recording

This sub block contains the general purpose acquisition parameters such as the scan field parameters, the scan modes and the start stop conditions.

Recording	
0x010000001 TYPE_ASCII	RECORDING_ENTRY_NAME The name of the recording as specified in the “Save” dialog box.
0x010000002 TYPE_ASCII	RECORDING_ENTRY_DESCRIPTION A short description which can be used to quickly select from a small number of recordings in a database. “Description” can be specified by the user in the “Save/Save As” dialog boxes or in the form-view of a database document.
0x010000003 TYPE_ASCII	RECORDING_ENTRY_NOTES Detailed description of a recording which can be specified by the user in the “Save/Save As” dialog boxes or in the form-view of a database document.
0x010000004 TYPE_ASCII	RECORDING_ENTRY_OBJECTIVE The name of the objective used during acquisition.
0x010000005 TYPE_ASCII	RECORDING_ENTRY_PROCESSING_SUMMARY A string that contains a short description of the image modifications that have been made to the image memory.
0x010000006 TYPE_ASCII	RECORDING_ENTRY_SPECIAL_SCAN_MODE Additional characterization of the scan mode for stack and z-scan modes. The string can be "NoSpecialMode", "FocusStep", "OnTheFly" or "ZScanner".
0x010000007 TYPE_ASCII	RECORDING_ENTRY_SCAN_TYPE Not used and should be an empty string.
0x010000008 TYPE_ASCII	RECORDING_ENTRY_SCAN_MODE A string that specifies scan mode and can be “Point”, “Line”, “Plane”, “ZScan”, “Stack”, “LineSelect” or “Imported File”. In the latter case it is unknown which scan mode was used.
0x010000009 TYPE_LONG	RECORDING_ENTRY_NUMBER_OF_STACKS The number of stacks within whole record (time-dimension).
0x01000000A TYPE_LONG	RECORDING_ENTRY_LINES_PER_PLANE The number of scan lines within each scan plane (y-dimension).
0x01000000B TYPE_LONG	RECORDING_ENTRY_SAMPLES_PER_LINE The number of samples within each scan line (x-dimension).
0x01000000C TYPE_LONG	RECORDING_ENTRY_PLANES_PER_VOLUME The number of scan planes within each stack (z-dimension).
0x01000000D TYPE_LONG	RECORDING_ENTRY_IMAGES_WIDTH Size of the image stored in the file x-direction.
0x01000000E TYPE_LONG	RECORDING_ENTRY_IMAGES_HEIGHT Size of the image stored in the file y-direction.
0x01000000F TYPE_LONG	RECORDING_ENTRY_IMAGES_NUMBER_PLANES Size of the images stored in the file z-direction.
0x010000010 TYPE_LONG	RECORDING_ENTRY_IMAGES_NUMBER_STACKS Size of the images stored in the file time-direction.

Recording 2nd page	
0x010000011 TYPE_LONG	RECORDING_ENTRY_IMAGES_NUMBER_CHANNELS Size of the images stored in the file channel-direction.
0x010000012 TYPE_LONG	RECORDING_ENTRY_LINSCAN_XY_SIZE Size of the background image in x- and y-direction. For line scan line selection there is a secondary image (background image) stored in the file.
0x010000013 TYPE_LONG	RECORDING_ENTRY_SCAN_DIRECTION Bi-directional scan.1 for bi-directional scan, 0 for unidirectional scan.
0x010000014 TYPE_LONG	RECORDING_ENTRY_TIME_SERIES Time series flag. 1 for time series, 0 if not a time series.
0x010000015 TYPE_LONG	RECORDING_ENTRY_ORIGINAL_SCAN_DATA Information if the scan memory contains original scan data. 1 for original data. 0 for calculated data.
0x010000016 TYPE_RATIONAL	RECORDING_ENTRY_ZOOM_X The x-scanner zoom factor in the range 0.7...40.
0x010000017 TYPE_RATIONAL	RECORDING_ENTRY_ZOOM_Y The y-scanner zoom factor in the range 0.7...40.
0x010000018 TYPE_RATIONAL	RECORDING_ENTRY_ZOOM_Z Not used and should be 1.0.
0x010000019 TYPE_RATIONAL	RECORDING_ENTRY_SAMPLE_0X X-scanner-offset of the center of the scanned image relative to the center for the maximum possible scan field in microns.
0x01000001A TYPE_RATIONAL	RECORDING_ENTRY_SAMPLE_0Y Y-scanner-offset of the center of the scanned image relative to the center for the maximum possible scan field in microns.
0x01000001B TYPE_RATIONAL	RECORDING_ENTRY_SAMPLE_0Z Z-offset of the first slice for “Stacks” or first line for “z-Scans” relative to the position of the focus drive prior the start of the acquisition.
0x01000001C TYPE_RATIONAL	RECORDING_ENTRY_SAMPLE_SPACING Center to center distance of adjacent pixels in x-direction in microns.
0x01000001D TYPE_RATIONAL	RECORDING_ENTRY_LINE_SPACING Center to center distance of adjacent pixels in y-direction in microns.
0x01000001E TYPE_RATIONAL	RECORDING_ENTRY_PLANE_SPACING Center to center distance of adjacent pixels in z-direction in microns.
0x01000001F TYPE_RATIONAL	RECORDING_ENTRY_PLANE_WIDTH The width of the scan field (x-direction) in microns.
0x010000020 TYPE_RATIONAL	RECORDING_ENTRY_PLANE_HEIGHT The height of the scan field (x-direction) in microns.
0x010000021 TYPE_RATIONAL	RECORDING_ENTRY_VOLUME_DEPTH The depth of the scan field (z-direction) in microns.

Recording 3rd page	
0x010000034 TYPE_RATIONAL	RECORDING_ENTRY_ROTATION The “Euler” rotation angle of the scan coordinates relative to the microscope coordinates in degrees. This value is used for rotated scans.
0x010000035 TYPE_RATIONAL	RECORDING_ENTRY_PRECESSION Not used and should be 0.0.
0x010000036 TYPE_RATIONAL	RECORDING_ENTRY_SAMPLE_0TIME The time at which the first pixel was acquired. The value is given in days relative to December 30, 1899.
0x010000037 TYPE_ASCII	RECORDING_ENTRY_START_SCAN_TRIGGER_IN The trigger in signal of the user port of the electronic unit that was used to start the scan operation and can be “Trigger1”, “Trigger2”, “Trigger3”, “Trigger4” or an empty string. The string is not used if the time series start signal (in entry 0x010000039) is not 1, 2 or 3 (trigger) or if it is not a time series.
0x010000038 TYPE_ASCII	RECORDING_ENTRY_START_SCAN_TRIGGER_OUT The trigger out signal at the user port of the electronic unit that was configured to signal the start of the scan operation and can be “Trigger1”, “Trigger2”, “Trigger3”, “Trigger4” or an empty string. The string is not used if it is not a time series.
0x010000039 TYPE_LONG	RECORDING_ENTRY_START_SCAN_EVENT The event that was configured to start a time series: 0 - Button (normal operation), 1 - Trigger slow - Scanner off, PMT high voltage off, reaction 300 msec, 2 - Trigger normal – Scanner off, PMT high voltage on, reaction 30 msec, 3 - Trigger fast - Scanner on, PMT high voltage on, reaction 5 msec, 4 - Start time.
0x010000040 TYPE_RATIONAL	RECORDING_ENTRY_START_SCAN_TIME The time when the scan operation was configured to be started. The value is given in days. The fractional part is used only. The value is not used if the start event (in entry 0x010000039) is not 4.
0x010000041 TYPE_ASCII	RECORDING_ENTRY_STOP_SCAN_TRIGGER_IN The trigger in signal of the user port of the electronic unit that was configured for stop of the scan operation and can be “Trigger1”, “Trigger2”, “Trigger3”, “Trigger4” or an empty string. The string is not used if the time series start signal (in entry 0x010000043) is not 1 (trigger) or if it is not a time series.
0x010000042 TYPE_ASCII	RECORDING_ENTRY_STOP_SCAN_TRIGGER_OUT The trigger out signal at the user port of the electronic unit that was configured to signal the end of the scan operation and can be “Trigger1”, “Trigger2”, “Trigger3”, “Trigger4” or an empty string. The string is not used if it is not a time series.
0x010000043 TYPE_LONG	RECORDING_ENTRY_STOP_SCAN_EVENT The event that was configured to stop a time series: 0 - Button (normal operation), 1 -Trigger 2 - Stop time.
0x010000044 TYPE_RATIONAL	RECORDING_ENTRY_START_SCAN_TIME The time when the scan operation was configured to be stopped. The value is given in days. The fractional part is used only. The value is not used if the stop event (in entry 0x010000043) is not 2.

Recording 4th page	
0x010000045 TYPE_LONG	RECORDING_ENTRY_USE_ROIS This value is different from 0 if ROIs have been used during acquisition.
0x010000046 TYPE_RATIONAL	RECORDING_ENTRY_USE_REDUCED_MEMORY_ROIS This value is different from 0 if image memory has been provided for the ROI bounding rectangle only.
0x010000047 TYPE_ASCII	RECORDING_ENTRY_USER The Windows NT user name of the user that has recorded the image. The entry is present in image files generated with release 2.8 or later.
0x010000048 TYPE_LONG	RECORDING_ENTRY_USEBCCORRECTION This value is different from 0 if the brightness and contrast correction for stacks and z-scans was active during acquisition.
0x010000049 TYPE_RATIONAL	RECORDING_ENTRY_POSITIONBCCORRECTION1 The brightness and contrast correction for stacks and z-scans is defined by two sets of values for AOTF power, PMT gain and detector amplifier gain and offset for at two positions of the focus drive. The values in entry 0x010000049 and 0x010000050 specify the two focus drive positions relative to the position of the focus dive at the start of the acquisition in microns.
0x010000050 TYPE_RATIONAL	RECORDING_ENTRY_POSITIONBCCORRECTION2 See entry 0x010000049.
0x010000051 TYPE_LONG	RECORDING_ENTRY_INTERPOLATIONY To achieve faster acquisition the scanning of every n-th scan line is possible. The data of the lines between the scanned lines are interpolated. The value in this entry specifies for how many image lines one line is scanned. The value “1” is used if no interpolation is done. The entry is present in files generated with release 3.0 or later.
0x010000052 TYPE_LONG	RECORDING_ENTRY_CAMERA_BINNING The binning factor that has been used to acquire the image with a CCD-Camera. If the image has not been acquired with a camera the value has no meaning. A binning factor n means that n horizontal and n vertical pixels of the CCD have been bined.
0x010000053 TYPE_LONG	RECORDING_ENTRY_CAMERA_SUPERSAMPLING The mode that was used to acuire an image that is larger than the size of the CCD-Camera. If the image has not been acquired with a camera the value has no meaning. The value 0 indicates that there was no technique use to enlarge the number of pixels. For the AxioCam color cameras the standard resolution is achieved without movement of the CCD by calculation of the color values for the individual pixels. The value 1 indicates also that there was no technique use to enlarge the number of pixels. For the AxioCam color cameras the standard resolution is achieved with movement of the CCD and aquisition of multiple images. Values greater than 1 indicate that the CCD was moved horizontally and vertically to achieve an horizontal and vertical enlargement of the number of pixels by this factor.

Recording 5th page	
0x010000054 0x010000055 TYPE_LONG	RECORDING_ENTRY_CAMERA_FRAME_WIDTH RECORDING_ENTRY_CAMERA_FRAME_HEIGHT The horizontal and vertical number of pixels that have been used to acuire an image that with the CCD-Camera. If the image has not been acquired with a camera the value has no meaning.
0x010000056 0x010000057 TYPE_RATIONAL	RECORDING_ENTRY_CAMERA_OFFSETX RECORDING_ENTRY_CAMERA_OFFSETY The horizontal and vertical offset of the center of the acuired image relative to the center of the CCD. If the image has not been acquired with a camera the value has no meaning.
0x010000059 TYPE_LONG	RECORDING_ENTRY_RT_BINNING The number of adjacent horizontal pixels that are combined during acquisition with the LSM 5 LIVE. A value n indicates that n-pixels are combined horizontally.
0x01000005a 0x01000005b TYPE_LONG	RECORDING_ENTRY_RT_FRAME_WIDTH RECORDING_ENTRY_RT_FRAME_HEIGHT The horizontal and vertical number of pixels of the mode for acquisition of an image with the LSM 5 LIVE. The *RECORDING_ENTRY_RT_BINNING*, *RECORDING_ENTRY_RT_FRAME_WIDTH* and *RECORDING_ENTRY_RT_FRAME_HEIGHT* properties are used by the acquisition controller to determine the LSM 5 LIVE image size mode. The acquisition controller uses the closest match to one of the pre-defined modes when there is no exact match.
0x01000005c 0x01000005d TYPE_LONG	RECORDING_ENTRY_RT_REGION_WIDTH RECORDING_ENTRY_RT_REGION_HEIGHT The horizontal and vertical number of pixels for acquisition of an image with the LSM 5 LIVE scanner. This can be a sub-region of the frame defined by the *RECORDING_ENTRY_RT_FRAME_WIDTH* and *RECORDING_ENTRY_RT_FRAME_HEIGHT* entries.
0x01000005e 0x01000005f TYPE_RATIONAL	RECORDING_ENTRY_RT_OFFSETX RECORDING_ENTRY_RT_OFFSETY The horizontal and vertical offset of the center of the acquisition region (with size secified by the *RECORDING_ENTRY_RT_REGION_WIDTH* and *RECORDING_ENTRY_RT_REGION_HEIGHT* properties) for the LSM 5 LIVE relative to the optical axis.
0x010000060 TYPE_RATIONAL	RECORDING_ENTRY_RT_ZOOM The zoom factor for the zoom optics of the LSM 5 LIVE.
0x010000061 TYPE_RATIONAL	RECORDING_ENTRY_RT_LINEPERIOD The time between the grabbing of two image lines with the LSM 5 LIVE.
0x010000062 TYPE_LONG	RECORDING_ENTRY_PRESCAN The value 1 is used if a time-series is started with a continuous acquisition to the first time index of the image. The user can then activate the actual time-series with a "TimeSeriesGo" command. The value is 0 if this mode has not been used.
0x010000063 TYPE_LONG	RECORDING_ENTRY_SCAN_DIRECTIONZ The value 1 indicates that the z-scanner (piezo objective or piezo stage) has been controlled in bi-directional mode. The value 0 is used when the z-scanner has been controlled in unidirectional mode.

9.2. Track

For one recording there can be more than one setting of the LSM device. These different setting includes:
- different illumination,
- different active detectors, detector voltage and amplifier settings,
- different beam splitters,
- different emission filters and
- different positions for the collimators.

The settings can be switched after a scan line or a frame. A “track” is such a device setting for simultaneously scanned channels.

For time series there can be an additional block of type “SUBBLOCK_TRACK“ for the settings used during bleach. No detectors are marked active in this track.

In addition, the online calculation of ratios is possible. The ratio settings are written to an additional block of type “SUBBLOCK_TRACK“. The “Track” entry for ratios does not contain device settings.

Track	
0x040000001 TYPE_LONG	TRACK_ENTRY_MULTIPLEX_TYPE Specifies when a switch to the next track is done. Possible values are: 1 - after a frame, 0 - after a line. The value is the same for all tracks in a recording.
0x040000002 TYPE_LONG	TRACK_ENTRY_MULTIPLEX_ORDER The one based index for the switch order of the tracks. It is possible that the blocks for the tracks are not arranged in ascending order.
0x040000003 TYPE_LONG	TRACK_ENTRY_SAMPLING_MODE Describes which method was used for the scan value generation: 0 - Sample The values are generated by integration over a time (“SampleObservationTime”) without average. 1 – Line average Same as “Sample” but with line average. “ 2 – Frame average Same as “Sample” but with frame average..
0x040000004 TYPE_LONG	TRACK_ENTRY_SAMPLING_METHOD The method which was used to calculate the scan values if the value in entry 040000005 is not 1. 1 - Mean, 2 - Sum.
0x040000005 TYPE_LONG	TRACK_ENTRY_SAMPLING_NUMBER The number of pixels to average or to add.
0x040000006 TYPE_LONG	TYPE_LONG TRACK_ENTRY_ACQUIRE The value 0 indicates that the track was not used during acquisition.
0x040000007 TYPE_RATIONAL	TRACK_ENTRY_SAMPLE_OBSERVATION_TIME The integration time in microseconds. For release 3.0 the entry is provided for compatibility reasons. For release 3.0 files the entry 0x040000019 (pixel time) should be used instead.
0x04000000B TYPE_RATIONAL	TRACK_ENTRY_TIME_BETWEEN_STACKS The time difference between the start of two scan cycles in seconds.

Track (2nd page)	
0x04000000C TYPE_ASCII	TRACK_ENTRY_NAME The name of the track as specified by the user in the “Configuration” window of the user interface. The name is used to distinguish different tracks in the user interface. For the track with the bleach setting the name “Bleach” is used internally by the program. For the track with the ratio setting the name “RatioTrack” is used by the program.
0x04000000D TYPE_ASCII	TRACK_ENTRY_COLLIMATOR1_NAME The name of the first collimator in the system.
0x04000000E TYPE_LONG	TRACK_ENTRY_COLLIMATOR1_POSITION The position of the first collimator in the system.
0x04000000F TYPE_ASCII	TRACK_ENTRY_COLLIMATOR2_NAME The name of the second collimator in the system.
0x040000010 TYPE_ASCII	TRACK_ENTRY_COLLIMATOR2_POSITION The position of the second collimator in the system.
0x040000011 TYPE_LONG	TRACK_ENTRY_IS_BLEACH_TRACK A value different from 0 indicates that the track contains the settings for bleach and has no active detection channels.
0x040000012 TYPE_LONG	TRACK_ENTRY_IS_BLEACH_AFTER_SCAN_NUMBER This entry is used if entry 040000011 (bleach track) and entry 0x040000006 (acquire) are not 0. A value different from 0 indicates that the bleach action is done after the scan cycle number specified in entry 040000013 (bleach scan number). The value is the same for all scan tracks.
0x040000013 TYPE_LONG	TRACK_ENTRY_BLEACH_SCAN_NUMBER The number of the scan cycle after which a bleach action should take place.
0x040000014 TYPE_ASCII	TRACK_ENTRY_TRIGGER_IN Not used and should be an empty string.
0x040000015 TYPE_ASCII	TRACK_ENTRY_TRIGGER_OUT Not used and should be an empty string.
0x040000016 TYPE_LONG	TRACK_ENTRY_IS_RATIO_TRACK Not used and should be 0.
0x040000017 TYPE_LONG	TRACK_ENTRY_BLEACH_COUNT The number of the bleach cycles to perform.
0x040000018 TYPE_RATIONAL	TRACK_ENTRY_SPI_CENTER_WAVELENGTH For acquisition with the “Meta” detector this entry specifies the wavelength of the light that it projected to the gap between the photomultiplier channels 16 and 17 (the center of the detector) in nanometers. For single channel detector this value has no meaning. The entry is not present in release 2.8 and earlier.
0x040000019 TYPE_RATIONAL	TRACK_ENTRY_PIXEL_TIME The pixel time in microseconds. In all releases prior 3.0 the integration time is 70% of the pixel time. In release 3.0 the ratio depends on the scan speed. For lower speeds the ratio is greater to achieve better sensitivity. The entry is not present in release 2.8 and earlier.
0x040000020 TYPE_ASCII	TRACK_ENTRY_ID_CONDENSOR_FRONTLENS The name of the condensor frontlens switch on the microscope stand. An empty string indicates that the condensor frontlens switch is not controlled during data acquisition.

Track (3rd page)	
0x040000021 TYPE_LONG	TRACK_ENTRY_CONDENSOR_FRONTLENS The position of the condensor front lens switch that has been used during acquisition. The position value 1 indicates that the front lens was in the beam path. The position value 0 indicates that no lens was in the beam path.
0x040000022 TYPE_ASCII	TRACK_ENTRY_ID_FIELD_STOP The name of the filed stop servo in the transmission beam path of the microscope stand. An empty string indicates that the filed stop servo is not controlled during data acquisition.
0x040000023 TYPE_RATIONAL	TRACK_ENTRY_FIELD_STOP_VALUE The position the servo for the field stop in the microscope stand in the range 0..1.
0x040000024 TYPE_ASCII	TRACK_ENTRY_ID_CONDENSOR_APERTURE The name of the condensor apertue servo in the transmission beam path of the microscope stand. An empty string indicates that the condensor apertue servo is not controlled during data acquisition.
0x040000025 TYPE_RATIONAL	TRACK_ENTRY_CONDENSOR_APERTURE The numerical aperture of the condesor that has been in the beam path during acquisition with the specified track.
0x040000026 TYPE_ASCII	TRACK_ENTRY_ID_CONDENSOR_REVOLVER The name of the condensor revolver in the transmission beam path of the microscope stand. An empty string indicates that the condensor revolver is not controlled during data acquisition.
0x040000027 TYPE_ASCII	TRACK_ENTRY_CONDENSOR_FILTER The filter that has been selected in the condensor revolver during acquisition with the specified track.
0x040000028 TYPE_RATIONAL	TRACK_ENTRY_ID_TRANSMISSION_FILTER1 The name of the first filter revolver in the transmission beam path of the microscope stand. An empty string indicates that the transmission filter revolver is not controlled during data acquisition.
0x040000029 TYPE_ASCII	TRACK_ENTRY_ID_TRANSMISSION1 The transmission (0..1)of the filter in the first transmission revolver of the microscope stand.
0x040000030 TYPE_RATIONAL	TRACK_ENTRY_ID_TRANSMISSION_FILTER2 The name of the second filter revolver in the transmission beam path of the microscope stand. An empty string indicates that the transmission filter revolver is not controlled during data acquisition.
0x040000031 TYPE_ASCII	TRACK_ENTRY_ID_TRANSMISSION2 The transmission (0..1) of the filter in the second transmission revolver of the microscope stand.
0x040000032 TYPE_LONG	TRACK_ENTRY_REPEAT_BLEACH The number of timeseries indexes after that the bleach action is repeated. The value 0 indicates that only one bleach action is performed. A value greater than 0 has no effect for the timeseries indexes prior to the bleach start event.
0x040000033 TYPE_LONG	TRACK_ENTRY_ENABLE_SPOT_BLEACH_POS A flag that indicates wheter or not the bleach position and acquisition postion was different for a spot scan.for the case of acquisition mode.

Track (4th page)	
0x040000034 0x040000035 TYPE_RATIONAL	TRACK_ENTRY_SPOT_BLEACH_POSX TRACK_ENTRY_SPOT_BLEACH_POSY The position of the laser spot that has been used to bleach the sample in case of a spot scan with enabled different bleach and acquisition positions.
0x040000036 TYPE_RATIONAL	TRACK_ENTRY_BLEACH_POSITION_Z The position of the focus drive for the bleach action relative to the postion at the start of the scan for stacks and z-scans. For plane, line, spot mode and bleach without grab the bleach z-position is relative to the focus drive position at the start of the bleach action.
0x040000037 TYPE_ASCII	TRACK_ENTRY_ID_TUBELENS The name of the tube lens revolver. An empty string indicates that the tube lens revolver has not been controlled.
0x040000038 TYPE_ASCII	TRACK_ENTRY_ID_TUBELENS_POSITION The name of the position that has been selected in the tube lens revolver during acquisition with the track. An empty string indicates that the tube lens revolver has not been controlled.
0x040000039	TRACK_TRANSMITTED_LIGHT The intensity of the transmitted light lamp during acquisition in the range 0..1.
0x04000003a	TRACK_REFLECTED_LIGHT The intensity of the reflected light lamp during acquisition in the range 0..1.
0x04000003b	TRACK_SIMULTAN_GRAB_AND_BLEACH The value 1 indicates that a bleach action was done during data acquisition with a second scanner pair, otherwise the value 0 is used.
0x04000003c	TRACK_BLEACH_PIXEL_TIME The pixel period in seconds, which has been used during a bleach action to control the scanner.

9.3. Laser

The sub block contains the information which lasers were used during the scan operation and which laser power was selected (if the laser power is adjustable).

Laser	
0x050000001 TYPE_LONG	OLEDB_LASER_ENTRY_NAME Name of the laser used for the data acquisition. Depending on the device configuration the name of the laser is read from the system and documented here. The names could be, e.g.,: - NeNe1 - NeNe2 - Ar/Kr - Argon - Argon2 - Enterprise - YAG - Mai Tai - Titanium:Sapphire
0x050000002 TYPE_LONG	OLEDB_LASER_ENTRY_ACQUIRE A value different from 0 indicates that the laser was switched on.
0x050000003 TYPE_LONG	OLEDB_LASER_ENTRY_POWER Laser power in mW to be used for the scan data acquisition. If the power is not adjustable this value has no meaning.

9.4. Detection Channel

The sub block SUBBLOCK_DETECTION_CHANNEL contains the detector settings used during acquisition. The detector settings include the settings for point detectors (PMT), amplifiers, pinhole diameter and filters located immediately before the detector in the beam path.

Detection Channel	
0x070000003 TYPE_RATIONAL	DETCHANNEL_ENTRY_DETECTOR_GAIN PMT-voltage of the PMT specified in entry 0x07000000c(point detector name). For the monitor diode this value has no meaning.
0x070000005 TYPE_RATIONAL	DETCHANNEL_ENTRY_AMPLIFIER_GAIN Amplifier gain value for the amplifier used for data acquisition with this detection channel.
0x070000007 TYPE_RATIONAL	DETCHANNEL_ENTRY_AMPLIFIER_OFFS Amplifier offset value for the amplifier used for data acquisition with this detection channel.
0x070000009 TYPE_RATIONAL	DETCHANNEL_ENTRY_PINHOLE_DIAMETER Pinhole diameter for the pinhole used for data acquisition with this detection channel in microns.
0x07000000b TYPE_LONG	DETCHANNEL_ENTRY_ACQUIRE This value is different from 0 if the detection cannel was used during acquisition.
0x07000000c TYPE_ASCII	DETCHANNEL_POINT_DETECTOR_NAME Name of the point detector (PMT) used for the data acquisition. The used names are listed in the column “Photo Multiplier” of table 8..
0x07000000d TYPE_ASCII	DETCHANNEL_AMPLIFIER_NAME The data acquisition electronics (except for the meta detector) has 4 digitizer channels. Each digitizer channel has an amplifier with selectable gain and offset. PMTs and monitor diode must be connected via a multiplexer with these amplifiers. The entry contains the name of the amplifier that was used for the PMT (or monitor diode). Due to the more complex electronics for the meta detector the program assigns the amplifiers internally and the string is no longer used in release 3.0.
0x07000000e TYPE_ASCII	DETCHANNEL_PINHOLE_NAME Name of the pinhole used for the detector. The used names are listed in table 8.
0x07000000f TYPE_ASCII	DETCHANNEL_FILTER_SET_NAME Name of the filter set which is located immediately before the detector in the beam path. The used names are listed in table 8.
0x070000010 TYPE_ASCII	DETCHANNEL_FILTER_NAME Name of the filter which should be used for the filter set in entry 0x070000010 (filter set name).
0x070000013 TYPE_ASCII	DETCHANNEL_INTEGRATOR_NAME The name of the integrator for the point detector in entry 0x07000000c (Point detector name). The used names are listed in table 8.

Detection Channel (2nd page)	
0x070000014 TYPE_ASCII	DETCHANNEL_DETECTION_CHANNEL_NAME Name which identifies the detection channel. The possible names are listed in table 8. The name must be used to find out which data channel is connected to which detection channel.
0x070000015 TYPE_RATIONAL	DETCHANNEL_ENTRY_DETECTOR_GAIN_BC1 If the brightness and contrast correction for stacks and z-scans is enabled, the entries 0x070000015 to 0x070000019 and the entry 0x070000020 are pairs of settings for: PMT-voltage: DETCHANNEL_ENTRY_DETECTOR_GAIN_BC1 and DETCHANNEL_ENTRY_DETECTOR_GAIN_BC2 amplifier gain: DETCHANNEL_ENTRY_AMPLIFIER_GAIN_BC1 and DETCHANNEL_ENTRY_AMPLIFIER_GAIN_BC2 amplifier offset: DETCHANNEL_ENTRY_AMPLIFIER_OFFS_BC1 and DETCHANNEL_ENTRY_AMPLIFIER_OFFS_BC2 for the two positions in entry 0x010000049 and 0x010000050 of the Recording sub block. The actual for a particular height was calculated by linear interpolation.
0x070000016 TYPE_RATIONAL	DETCHANNEL_ENTRY_DETECTOR_GAIN_BC2 See entry 0x070000015.
0x070000017 TYPE_RATIONAL	DETCHANNEL_ENTRY_AMPLIFIER_GAIN_BC1 See entry 0x070000015.
0x070000018 TYPE_RATIONAL	DETCHANNEL_ENTRY_AMPLIFIER_GAIN_BC2 See entry 0x070000015.
0x070000019 TYPE_RATIONAL	DETCHANNEL_ENTRY_AMPLIFIER_OFFS_BC1 See entry 0x070000015.
0x070000020 TYPE_RATIONAL	DETCHANNEL_ENTRY_AMPLIFIER_OFFS_BC2 See entry 0x070000015.
0x070000021 TYPE_LONG	DETCHANNEL_ENTRY_SPECTRAL_SCAN_CHANNELS The number of image channels that have been generated for the spectral scan pseudo channel (name “ChS”) in the spectral scan mode. The entry is new for release 3.0 and has no meaning for all scan modes that are not spectral scans.
0x070000022 TYPE_RATIONAL	DETCHANNEL_ENTRY_SPI_WAVELENGTH_START The entries 0x070000022 and 0x070000023 contain the wavelength range in nanometers for detection with the “Meta” detector. The entry is used for the detectors with the names “ChS1” to “ChS8” and “ChS”. For all other detectors the entry has no meaning. The entry is new for release 3.0.
0x070000023 TYPE_RATIONAL	DETCHANNEL_ENTRY_SPI_WAVELENGTH_END See entry 0x070000022.
0x070000026 TYPE_ASCII	DETCHANNEL_ENTRY_DYE_NAME Name of the dye the user has selected for a “Meta” detector channel in the user interface. The entry is new for release 3.0.
0x070000027 TYPE_ASCII	DETCHANNEL_ENTRY_DYE_FOLDER Name of the dye folder the user has selected for a “Meta” detector channel in the user interface. The entry is new for release 3.0.

9.5. Illumination Channel

The sub block contains the settings for the attenuators in the excitation beam path. For most devices the attenuators are acousto optical tunable filters (AOTFs). For the LSM 5 PASCAL mechanical attenuators are used.

Illumination Channel	
0x090000001 TYPE_ASCII	ILLUMCHANNEL_ENTRY_NAME Name of the illumination channel. In older versions the string “AOTF” was used in all Illumination channels. In the newer versions a string with the wavelength in nm is used.
0x090000002 TYPE_RATIONAL	ILLUMCHANNEL_ENTRY_POWER Power of the light that transmits the attenuator in percent.
0x090000003 TYPE_RATIONAL	ILLUMCHANNEL_ENTRY_WAVELENGTH The wavelength of interest of the light source in nm.
0x090000004 TYPE_LONG	ILLUMCHANNEL_ENTRY_AQUIRE A value different from 0 indicates the attenuator was enabled. If an attenuator is disabled there is no transmission light for this wavelength.
0x090000005 TYPE_ASCII	ILLUMCHANNEL_DETCHANNEL_NAME For automatic collimator correction it is required to know which laser line is detected in which point detector. The entry specifies which detection channel will receive the light of this illumination channel. An empty string is used to indicate that the connection is unknown.
0x090000006 TYPE_RATIONAL	ILLUMCHANNEL_ENTRY_POWER_BC1 If the brightness and contrast correction for stacks and z-scans is enabled the entries 0x090000006 and 0x090000007 are the two values for the attenuator transmission at the two positions in entry 0x010000049 and 0x010000050 of the Recording sub block. The actual attenuator transmission value is calculated by linear interpolation.
0x090000007 TYPE_RATIONAL	ILLUMCHANNEL_ENTRY_POWER_BC2 See entry 0x090000006.

9.6. Beam Splitter

The sub block contains the setting for the dichroic beam splitters in the beam path.

Beam Splitter	
0x0b0000001 TYPE_ASCII	BEAMSPLITTER_ENTRY_FILTER_SET Identifier for the beam splitter. The name is set to the same string as entry 0x0b0000003.
0x0b0000002 TYPE_ASCII	BEAMSPLITTER_ENTRY_FILTER Name of the filter used in the filter set during acquisition.
0x0b0000003 TYPE_ASCII	BEAMSPLITTER_ENTRY_NAME The name of the filter set. The following strings are used: HT Main beam splitter. NT1 First dichroic beam splitter. NT2 Second dichroic beam splitter. NT3 Third dichroic beam splitter. FW1 Main beam splitter for non-descanned detection. NDD_NT1 Beam splitter in the reflection beam path for non-descanned detection. NDD_NT2 Beam splitter in the transmission beam path for non-descanned detection.

9.7. Data Channel

The sub block contains channel dependent settings for image memory, display and calculation of scan data during the scan operation.

Data Channel	
0x0d0000001 TYPE_ASCII	DATACHANNEL_ENTRY_NAME Identifier for the data channels. The name for scanned channels must be the same as the name in the corresponding entry 0x070000014 (detection channel name) in the SUBBLOCK_DETECTION_CHANNEL sub block for the detector that has been used to generate the data for the image channel (See table 8 for possible names). In addition, there are data channel entries for online ratio calculations with the names „R1“ (first ratio) and „R2“ (second ratio).
0x0d0000004 TYPE_LONG	DATACHANNEL_ENTRY_COLOR The color which was selected in the „Beam Path“-control for the channel with intensity values in the range 0..255 for the three color components. Msb Lsb 0 | blue | green | red
0x0d0000005 TYPE_LONG	DATACHANNEL_ENTRY_SAMPLETYPE the type of the acquired data (8 bit or 12 bit data). The following values are used 8 bit channel - value “1” 12 bit channel - value “2”.
0x0d0000006 TYPE_LONG	DATACHANNEL_ENTRY_BITSPERSAMPLE The number of bits used to store the pixel data for the data channel. The following values are used 8 bit channel - value “8” and 12 bit channel - value “12”. In all releases the value is the same for all channels for one recording.
0x0d0000007 TYPE_LONG	DATACHANNEL_ENTRY_RATIO_TYPE the type of an online calculation where this “data channel” is the destination. The following values are possible: 0 - no online calculation - the data channel receives raw scan data. 1 - online ratio: (S1+C1)/(S2+C2)*C3+C4 2 - online subtraction: (S1*C1-S2*C3)/C2+C4 3 - online ratio: (S1-S2*C1)/(S1+S2*C2)*C3+C4 4 - online hill function: C3*(C4/C5)*(S1-C1)/(C2-S2) 5 - online reference ratio: (S1-S2+C1)/(S2-C2)*C3+C4 6 - online linear unmixing The operands and constants are: S1 - Source1 S2 - Source2 C1 - RatioConstant1 C2 - RatioConstant2 C3 - RatioConstant3 C4 - RatioConstant4 C5 - RatioConstant5

Data Channel (2nd page)	
0x0d0000008 TYPE_LONG	DATACHANNEL_ENTRY_RATIO_TRACK1 The entries 0x0d0000008 to 0x0d000000b specify the two source operands for an online calculation if the ratio type in entry 0x0d0000007 is not 0. DATACHANNEL_ENTRY_RATIO_TRACK1 DATACHANNEL_ENTRY_RATIO_TRACK2 - Multiplex order of the track (see entry 0x040000002 in the track sub block). DATACHANNEL_ENTRY_RATIO_CHANNEL1 DATACHANNEL_ENTRY_RATIO_CHANNEL2 - Name of the detection channel.(see entry 0x070000014 in the detection channel sub block.
0x0d0000009 TYPE_LONG	DATACHANNEL_ENTRY_RATIO_TRACK2 See entry 0x0d0000008.
0x0d000000a TYPE_ASCII	DATACHANNEL_ENTRY_RATIO_CHANNEL1 See entry 0x0d0000008.
0x0d000000b TYPE_ASCII	DATACHANNEL_ENTRY_RATIO_CHANNEL2 See entry 0x0d0000008.
0x0d000000c 0x0d000000d 0x0d000000e 0x0d000000f 0x0d0000010 0x0d0000011 TYPE_RATIONAL	DATACHANNEL_ENTRY_RATIO_CONST1 DATACHANNEL_ENTRY_RATIO_CONST2 DATACHANNEL_ENTRY_RATIO_CONST3 DATACHANNEL_ENTRY_RATIO_CONST4 DATACHANNEL_ENTRY_RATIO_CONST5 DATACHANNEL_ENTRY_RATIO_CONST6 The entries 0x0d000000c to 0x0d000000f contain the six constants C1 to C6 for online ratio calculations (see entry 0x0d0000007).
0x0d0000012 0x0d0000013 TYPE_LONG	DATACHANNEL_ENTRY_RATIO_FIRST_IMAGES1 DATACHANNEL_ENTRY_RATIO_FIRST_IMAGES2 The number of the first cycles of a timeseries where the source data have been averaged and are used used as ratio source. If the value is zero the averaging of first images was off and the correponding images have been used as ratio source instead.
0x0d0000014 TYPE_ASCII	DATACHANNEL_ENTRY_DYE_NAME The name of the dye that the user has selected form the spectral database for an online linear unmixing calculation.
0x0d0000015 TYPE_ASCII	DATACHANNEL_ENTRY_DYE_FOLDER The name of the dye folder that the user has selected form the spectral database for an online linear unmixing calculation.
0x0d0000016 TYPE_ASCII	DATACHANNEL_ENTRY_SPECTRUM A string with the spectrum (intensity values only) that has been used for linear unmixing for the image channel.
0x0d0000017 TYPE_LONG	DATACHANNEL_ENTRY_ACQUIRE A flag that indicates whether or not the data have been stored to the image. If not, the data have been pure source data for ratios.

9.8. Timer

For time series it is possible to change the cycle delay during acquisition. The “Timer” sub block contains a set of cycle delays. The user can switch between these cycle delays during acquisition.

Timer	
0x012000001 TYPE_ASCII	TIMER_ENTRY_NAME Identifier for the timer. The strings “Timer1” to “Timer6” are used.
0x012000003 TYPE_RATIONAL	TIMER_ENTRY_INTERVAL Scan cycle delay or time interval in seconds. Depending on the user specific device setting this can be either the time difference between two starts of the scanning of stacks (time interval) or the time between end of the previous and start of the next stack (cycle delay).
0x012000004 TYPE_ASCII	TIMER_ENTRY_TRIGGER_IN Specifies which input signal of the user port of the electronic unit activates the cycle delay value in entry 0x012000003. The trigger signals “Trigger1”, “Trigger2”, “Trigger3” and “Trigger4” are supported. The entry can also contain an empty string or “None”.
0x012000005 TYPE_ASCII	TIMER_ENTRY_TRIGGER_OUT Specifies which output signal of the user port of the electronic unit should be set after the user has switched this timer. The trigger signals “Trigger1”, “Trigger2”, “Trigger3” and “Trigger4” are supported. The entry can also contain an empty string or “None”.

9.9. Marker

Markers are user defined character strings which are used during a scan operation to record user actions. The marker string is recorded with time and image number if the user activates the maker in the user interface or an event is signaled at the user port of the electronic unit.

Marker	
0x014000001 TYPE_ASCII	MARKER_ENTRY_NAME Identifier for the marker. The strings “Marker1” to “Marker 6” are used by the LSM program.
0x014000002 TYPE_ASCII	MARKER_ENTRY_DESCRIPTION A user specified string which is recorded with time and image number when the user or an external event has activated the marker.
0x014000003 TYPE_ASCII	MARKER_ENTRY_TRIGGER_IN Specifies which input of the user port of the electronic unit triggers the recording of the marker event. The trigger signals “Trigger1”, “Trigger2”, “Trigger3” and “Trigger4” are supported. The entry can also contain an empty string or “None”.
0x014000004 TYPE_ASCII	MARKER_ENTRY_TRIGGER_OUT Specifies which output of the user port of the electronic unit should be set after the user has manually signaled the marker event. The trigger signals “Trigger1”, “Trigger2”, “Trigger3” and “Trigger4” are supported The entry can also contain an empty string or “None”.

10. Channel Wavelength Range

For images acquired with the meta detector the wavelength range for the detected emission light is known. The information is stored in a block in the image file. The *u32OffsetChannelWavelength* entry of the CZ-Private tag contains the offset to this block.

SINT32	s32Channels	Number of channels for which wavelength information is stored.
FLOAT64	f64StartWavelength1	Start and end wavelength for the image channels in
FLOAT64	f64EndWavelength1	meter.
FLOAT64	f64StartWavelength2	
FLOAT64	f64EndWavelength2	
...		

11. Tile Positions

For images acquired in tile-scan mode the positions of the individual tiles (dimension M) are stored in a list in a block in the image file. The *u32OffsetTilePositions* entry of the CZ-Private tag contains the offset to this block. Depending on the acquisition mode the tile position are stored relative to the acquisition region or relative to the zero position of the stage.

UINT32	u32Tiles	The number of tiles for which position information is stored.
FLOAT64	f64TilePositionX1	The horizontal and vertical positions for the tiles in meter
FLOAT64	f64TilePositionY1	
FLOAT64	f64TilePositionZ1	
FLOAT64	f64TilePositionX2	
FLOAT64	f64TilePositionY2	
FLOAT64	f64TilePositionZ2	
...		

Note:
1) The Z- coordinate is not used in the .lsm format rel 6.2 or earlier. Zero (“0” is written to all “TilePositionZ”
2) all positions are only relative positions. There is no absolute motor positions in ZEN /.lsm file formats

12. Positions

For images acquired in multi-position mode the positions of the individual acquisitions (dimension P) are stored in a list in a block in the image file. The *u32OffsetPositions* entry of the CZ-Private tag contains the offset to this block.

UINT32	u32Positions	The number of acquisition regions for which position information is stored.
FLOAT64	f64PositionX1	The horizontal and vertical positions for the regions in meter
FLOAT64	f64PositionY1	
FLOAT64	f64PositionZ1	
FLOAT64	f64PositionX2	
FLOAT64	f64PositionY2	
FLOAT64	f64PositionZ2	
...		

Note:
1) This block (all position coordinates) are available only from rel 6.2 onwards.
2) all positions are only relative positions. There is no absolute motor positions in ZEN /.lsm file formats

Note:
For tile images acquired in multi-position mode, the coordinates of tile positions and “positions” have to be added:
TilePositionX = f64TilePositionX1 + f64PositionX1
TilePositionY = f64TilePositionY1 + f64PositionY1
TilePositionZ = f64PositionZ1
For Z, we store no TilePosition, as tiling is only possible in X-Y.

13. Files larger than 4 GB

The TIFF specification limits the maximum file size to 4 GB due to the usage of 32 bit file offsets. To overcome this limitation the LSM 5/7 program is using a wrap around mechanism.

The writer truncates 64 bit file offsets to 32 bit.

For files larger than 4 GB all data (directories and meta-data) except image and thumbnail pixels are always written within the first 4 GB. A special treatment by the reader is limited to the access of image pixels and thumbnail pixels.

The content of the tables with the strip offsets can contain truncated file offsets. The writer ensures that the file offsets are assigned ascending in the order they appear in the TIFF image directories.

To read the image pixels the reader checks for file offset truncation and reconstructs the 64-bit file offsets by comparing a new file offset with the previous one. In this mechanism a 64 bit variable *Wrap* is initialized with 0 at the start. A 32-bit variable *PreviousOffset* is used to remember a previous read file offset and is also initialized with 0. A third 32-bit variable *CurrentOffset* is declared.

The reader iterates through the entries in the strip offset tables in the order they appear in the image directories. For each file offset the following steps are performed:

1) The 32-bit file offset for the image pixel region form the image is written to the variable *CurrentOffset.*
2) If *CurrentOffset* is smaller than *PreviousOffset* then the value 2 by the power of 32 is added to the 64-bit variable *Wrap.*
3) The image pixels are accessed at the file offset *CurrentOffset* + *Wrap.*
4) The value *CurrentOffset* is assigned to the variable *PreviousOffset,*

This sequence is repeated until all image pixels have been read.

14. ApplicationTags / KS data

The Zeiss Vision KS100 to KS400 programs, as well as the AxioVision program, can write data to this field. It can also be used to store generic image information, e.g. by custom macros. The *u32OffsetKsData* entry of the CZ-Private tag contains the offset to this block, and the offset is 0 if this block is not present.

UINT32	u32BlockSize	The size of this block (in Byte)
SINT32	s32NumEntries	Number of entries
UINT32	u32EntrySize1	The size of this entry (in Byte)
SINT32	s32EntryNameLength1	Length of the entry name string
STRING	sEntryName1	Entry name, '\0'-terminated ANSI string
SINT32	s32DataType1	Type 2 – string 4 – int32 5 – float64 6 - date 7 - bool
SINT32	s32DataSize1	Size of data block: Bool, int32: 4 Float64, date: 8 String: length of ANSI string
SINT32 FLOAT64 STRING	Data1	Data block of variable length. Length depends on s32DataSize1 field.
SINT32	s32ImageDimensions1	Number of dimensions in an image. Can be missing, together with the s32Coordinate1x values.
SINT32	s32Coordinate11	Coordinate of application tag in first dimension. If negative, tag applies to all coordinates in this dimension. Can be missing, together with s32ImageDimensions1.
...		…
SINT32	s32Coordinate1N	Coordinate of application tag in highest dimension. If negative, tag applies to all coordinates in this dimension. Can be missing, together with s32ImageDimensions1.
UINT32	u32EntrySize2	The size of this entry
…		…
SINT32	s32Coordinate2N	Coordinate of application tag in highest dimension. If negative, tag applies to all coordinates in this dimension. Can be missing, together with s32ImageDimensions2.
UINT32	u32EntrySize3	The size of this entry
...		…