OpenOffice.org Source Project

OpenOffice.org's Documentation of the

Microsoft® Excel File Format

Excel Versions 2, 3, 4, 5, 95, 97, 2000, XP

Author	Daniel Rentz <daniel.rentz@sun.com>

Source	PDF: http://sc.openoffice.org/excelfileformat.pdf XML: http://sc.openoffice.org/excelfileformat.sxw
Project started	2001-Jun-29
Last change	2001-Nov-24

Contents

1 Introduction

1.1 File Format Versions

The Excel file format is named BIFF (Binary Interchange File Format). The following table shows which Excel version writes which file format.

Excel version	BIFF version	Document type	File format
Excel 2	BIFF2	Worksheet	Stream
Excel 3	BIFF3	Worksheet	Stream
Excel 4	BIFF4	Worksheet or workbook	Stream
Excel 5.0	BIFF5	Workbook	OLE2 storage
Excel 7.0 (Excel 95)	BIFF7	Workbook	OLE2 storage
Excel 97, 2000, XP	BIFF8	Workbook	OLE2 storage

The oldest file format BIFF2 has of course the most restrictions. From BIFF4 on it is possible to store a bundle of sheets, called a workbook. The current format BIFF8 contains major changes towards older BIFF versions, for instance the handling of Unicode strings.

1.2 Structure of a Worksheet File (BIFF2-BIFF4)

Files stored in the BIFF versions BIFF2 to BIFF4 contain all records for a sheet or a BIFF4 workbook in one simple stream. The records are arranged sequential, they are never embedded in other records.

1.3 Structure of a Workbook File (BIFF5-BIFF8)

An Excel workbook with several sheets (from BIFF5 on) is stored as an OLE2 compound file. It contains several streams for different types of data. The following table lists names of possible streams.

Stream name	Contents
Book	BIFF5/BIFF7 workbook stream (→4.3)
Workbook	BIFF8 workbook stream (→4.3)
<05$_H$>SummaryInformation	Document settings
<05$_H$>DocumentSummaryInformation	Document settings
User Names	User names in shared workbooks (→9)
Revision Log	Change tracking log stream (→9)

The names of the streams SummaryInformation and DocumentSummaryInformation contain a leading byte with the value 05_H.

It is possible to create substorages like subdirectories in a file system, for instance for the pivot table streams. These storages contain substreams itself.

Storage name	Contents
LNKxxxxxxxx	Storage for a linked OLE object (→6)
MBDxxxxxxxx	Storage for an embedded OLE object (→6)
_SX_DB_CUR	Pivot cache storage. The streams contain cached values for one or more PivotTables (→8).
_VBA_PROJECT_CUR	Visual BASIC project storage

In all streams the records are arranged sequential, they are never embedded in other records. Exception in BIFF8: The Escher object stream is splitted and embedded in several MSODRAWING records (→6).

1.4 Structure of a Record

In an Excel data stream the data is divided into several records. Each record contains specific data for the various features of Excel. The common structure of a record is described in the following table.

Offset	Size	Contents	
0	2	Identifier	} Record header
2	2	Size of the following data (sz)	
4	sz	Data	

The maximum size of the record data is limited. If the size of the record data exceeds the given limits, one or more CONTINUE (→5.6) records will be added. Inside of a CONTINUE record the data of the previous record continues as usual.

In the following descriptions only the record data without the headers is shown. All offsets are relative to the beginning of the record data and not to the entire record. The contents of most of the records differ from version to version. This will be described in separate tables. A few older records are replaced in newer BIFF versions. Excel does not write these old records anymore, but can still read them.

1.5 Byte Order

All data items containing more than one byte are stored using the Little-Endian method. That means the least significant byte is stored first and the most significant byte last. This is valid for all data types like 16-bit-integers, 32-bit-integers, floating-point values and Unicode characters. For instance the 16-bit-integer value 1234_H is converted into the byte sequence $34_H\ 12_H$.

2 Basic Substructures

This chapter contains information about substructures which are part of several records, for instance strings or error codes.

2.1 Byte Strings (BIFF2-BIFF7)

All Excel file formats up to BIFF7 contain simple byte strings. The byte string consists of the length of the string followed by the character array. The length is stored either as 8-bit-integer or as 16-bit-integer, depending on the current record. The string is not zero-terminated.

Offset	Size	Contents
0	1 or 2	Length of the string (character count) (ln)
1 or 2	ln	Character array (8-bit-characters)

2.2 Unicode Strings (BIFF8)

From BIFF8 on, strings are stored in a new Unicode format which allows reading and writing 16-bit-characters. The following tables describe the standard format, but in many records the strings differ from this format. This will be mentioned separately. It is possible (but not required) to store Rich-Text formatting information and extended information for Far-East inside of an Unicode string. This results in four different ways to store a string. The string is not zero-terminated.

2.2.1 Contents of an Unicode string

The string consists of the character count (as usual an 8-bit-integer or a 16-bit-integer), option flags, the character array and optional formatting information. If the string is empty, sometimes the option flags field will not occur. This is mentioned at the respective place.

• Unicode string without additional information

Offset	Size	Contents
0	1 or 2	Length of the string (character count) (ln)
1 or 2	1	Option flags (see below): 00_H or 01_H
2 or 3	ln or 2·ln	Character array (8-bit-characters or 16-bit-characters)

• Unicode string with Rich-Text formatting information

Offset	Size	Contents
0	1 or 2	Length of the string (character count) (ln)
1 or 2	1	Option flags (see below): 08_H or 09_H
2 or 3	2	Number of Rich-Text formatting runs (rt)
4 or 5	ln or 2·ln	Character array (8-bit-characters or 16-bit-characters)
var.	4·rt	List of rt formatting runs. Each run contains two 16-bit indexes:

Offset	Size	Contents
0	2	First formatted character (zero-based)
2	2	Index to FONT record (→5.15)

• **Unicode string with Far-East information**

Offset	Size	Contents
0	1 or 2	Length of the string (character count) (ln)
1 or 2	1	Option flags (see below): 04_H or 05_H
2 or 3	4	Far-East data size (sz)
6 or 7	ln or 2·ln	Character array (8-bit-characters or 16-bit-characters)
var.	sz	Unknown extended data about phonetic, keyboard, etc.

• **Unicode string with Rich-Text and Far-East information**

Offset	Size	Contents
0	1 or 2	Length of the string (character count) (ln)
1 or 2	1	Option flags (see below): $0C_H$ or $0D_H$
2 or 3	2	Number of Rich-Text formatting runs (rt)
4 or 5	4	Far-East data size (sz)
8 or 9	ln or 2·ln	Character array (8-bit-characters or 16-bit-characters)
var.	4·rt	List of rt formatting runs. See above for details.
var.	sz	Unknown extended data about phonetic, keyboard, etc.

2.2.2 Option flags

Bit	Mask	Contents	
0	01_H	0 = 8-bit-characters	1 = 16-bit-characters
2	04_H	0 = Contains no Far-East info	1 = Contains Far-East info
3	08_H	0 = Contains no Rich-Text info	1 = Contains Rich-Text info

2.3 RK Values

An RK value is an encoded integer or floating-point value. RK values have a size of 4 bytes and are used to decrease file size for floating-point values.

Structure of an RK value (32-bit-value):

Bit	Mask	Contents	
0	00000001_H	0 = Value not changed	1 = Value multiplied by 100
1	00000002_H	0 = IEEE floating-point value	1 = Integer value
31-2	$FFFFFFFC_H$	Encoded value	

If bit 1 is set to 0, the encoded value represents the 30 most significant bits of an IEEE floating-point value. The 34 least significant bits must be set to zero. If bit 1 is set to 1, the encoded value represents a signed 30-bit-integer value.

If bit 0 is set to 1, the decoded value must be divided by 100 to get the final result.

Examples:

RK value	Decoded value	Result
$3FF00000_H$	Floating-point = 1	1
$3FF00001_H$	Floating-point = 1	0.01
$004B5646_H$	Integer = 1234321	1234321
$004B5647_H$	Integer = 1234321	12343.21

2.4 Error Codes

If the calculation of a formula results in an error or any other action fails, Excel sets a specific error code. These error codes are used for instance in cell records and formulas.

Error code	Error value	Description
00_H	#NULL!	Intersection of two cell ranges is empty
07_H	#DIV/0!	Division by zero
$0F_H$	#VALUE!	Wrong type of operand
17_H	#REF!	Illegal or deleted cell reference
$1D_H$	#NAME?	Wrong function or range name
24_H	#NUM!	Value range overflow
$2A_H$	#N/A!	Argument or function not available

2.5 List of Cached Values

The records CRN (→5.7) and EXTERNNAME (→5.12) and the formula token ptgArray (array constant, →3.10.1) require a list of constant values (floating-point values, strings, boolean values or error codes). These values are stored as a simple list. The number of values is stored before in the respective record or token.

• IEEE floating-point value

Offset	Size	Contents
0	1	01_H (identifier for a floating-point constant)
1	8	IEEE floating-point value

• String value

A string value, BIFF2-BIFF7:

Offset	Size	Contents
0	1	02_H (identifier for a string constant)
1	var.	Byte string, 8-bit string length (→2.1)

A string value, BIFF8:

Offset	Size	Contents
0	1	02_H (identifier for a string constant)
1	var.	Unicode string, 16-bit string length, option flags occur always (→2.2)

• Boolean value

Offset	Size	Contents
0	1	04_H (identifier for a boolean constant)
1	1	0 = FALSE, 1 = TRUE
2	7	Not used

• Error value

Offset	Size	Contents
0	1	10_H (identifier for an error constant)
1	1	Error code (→2.4)
2	7	Not used

2.6 Encoded Document Names

2.6.1 Encoded file names

The intention of encoding file names is to make them more platform independent. Encoded file names occur in the records EXTERNSHEET (BIFF2-BIFF7, →5.13) or SUPBOOK (BIFF8, →5.35) and DCONREF (→5.8).

The first character of the file name is used to determine the type of encoding. In Unicode strings (BIFF8) this could be a 16-bit-value.

First character	Meaning
00_H	Empty sheet name (nothing will follow)
01_H	Encoded file name
02_H	External reference to the own document (nothing will follow)
03_H	External reference to a sheet in the own document (BIFF5/BIFF7)
others	Not encoded. This is the first character of the file name.

Inside of the encoded file name there can occur several control characters.

Control character	Meaning
01_H	An MS-DOS drive letter will follow or „@“ for a local network path
02_H	Start path name on same drive as own document
03_H	End of subdirectory name
04_H	Start path name in parent directory of own document (may occur repeatedly)
06_H	Start path name in startup directory of Excel
09_H	Sheet in the same workbook (BIFF4)

Example: Own document is saved as „C:\path\own.xls“.

Formula	Encoded filename
`=own.xls!A1`	<02_H>
`=Sheet2!A1`	<01_H><09_H>Sheet2 (BIFF4 workbook)
`=Sheet2!A1`	<03_H>Sheet2 (BIFF5/BIFF7)
`=[ext.xls]Sheet1!A1`	<01_H>[ext.xls]Sheet1
`='sub\[ext.xls]'Sheet1!A1`	<01_H>sub<03_H>[ext.xls]Sheet1
`='\[ext.xls]'Sheet1!A1`	<01_H><02_H>[ext.xls]Sheet1
`='\sub\[ext.xls]'Sheet1!A1`	<01_H><02_H>sub<03_H>[ext.xls]Sheet1
`='\sub\sub2\[ext.xls]'Sheet1!A1`	<01_H><02_H>sub<03_H>sub2<03_H>[ext.xls]Sheet1
`='D:\sub\[ext.xls]'Sheet1!A1`	<01_H><01_H>Dsub<03_H>[ext.xls]Sheet1
`='..\sub\[ext.xls]'Sheet1!A1`	<01_H><04_H>sub<03_H>[ext.xls]Sheet1
`='\\pc\sub\[ext.xls]'Sheet1!A1`	<01_H><01_H>@pc<03_H>sub<03_H>[ext.xls]Sheet1

2.6.2 Encoded document names for DDE and OLE object links

A DDE link contains the name of the server application and the name of a document. An OLE object link contains a class name and a document name. In both cases the names are stored in one string, separated by the control character 03_H.

Example: A document contains a DDE link to the SO/OOo Calc document „example.sxc“ and an OLE object link to the bitmap file „example.bmp“.

Link	Encoded document name
DDE	soffice<03_H>example.sxc
OLE object	Package<03_H>example.bmp

2.7 Line Styles for Cell Borders (BIFF3-BIFF8)

These line styles are used to define cell borders. The styles 08_H to $0D_H$ are available in BIFF8 only.

Index	Style	Sample	Index	Style	Sample
00_H	No line			The following for BIFF8 only:	
01_H	Thin		08_H	Medium dashed	
02_H	Medium		09_H	Thin dash-dotted	
03_H	Dashed		$0A_H$	Medium dash-dotted	
04_H	Dotted		$0B_H$	Thin dash-dot-dotted	
05_H	Thick		$0C_H$	Medium dash-dot-dotted	
06_H	Double		$0D_H$	Slanted medium dash-dotted	
07_H	Hair				

2.8 Patterns for Cell Background Area (BIFF3-BIFF8)

From BIFF3 on, the cell background area may contain a pattern. Foreground and background colors of the pattern are defined separately. In the following table black is used as foreground color and white as background color.

Index	Pattern	Sample	Index	Pattern	Sample
00_H		No background			
01_H			$0A_H$		
02_H			$0B_H$		
03_H			$0C_H$		
04_H			$0D_H$		
05_H			$0E_H$		
06_H			$0F_H$		
07_H			10_H		
08_H			11_H		
09_H			12_H		

2.9 Cell Attributes (BIFF2)

All cell records in BIFF2 contain a cell attribute field with a size of 3 bytes. They contain an index to an XF record (→5.37) and some repeated contents of the referenced XF record. The XF index field has a size of only 6 bits, so the index range is 0-63. If an index >62 is used, the XF index field always contains the vaue 63, and an IXFE record (→5.20) occurs in front of the cell record. It contains the correct index of the XF record.

Cell attributes field (3 bytes), BIFF2:

Offset	Size	Contents		
0	1	Cell protection and XF index:		
		Bit	**Mask**	**Contents**
		5-0	$3F_H$	Index to XF record (→5.37). The value $3F_H$ (63) indicates a preceding IXFE record (→5.20).
		6	40_H	1 = Cell is locked
		7	80_H	1 = Formula is hidden
1	1	Indexes to FORMAT and FONT records:		
		Bit	**Mask**	**Contents**
		5-0	$3F_H$	Index to FORMAT record (→5.16)
		7-6	$C0_H$	Index to FONT record (→5.15)
2	1	Cell style:		
		Bit	**Mask**	**Contents**
		2-0	07_H	XF_HOR_ALIGN – Horizontal alignment (→5.37.1)
		3	08_H	1 = Cell has left black border
		4	10_H	1 = Cell has right black border
		5	20_H	1 = Cell has top black border
		6	40_H	1 = Cell has bottom black border
		7	80_H	1 = Cell has shaded background

3 Formulas

3.1 Common Structure

Formulas are stored as part of a record, for instance inside of a FORMULA record or a NAME record. The common format of a formula is as follows:

Formula in BIFF2:

Offset	Size	Contents
0	1	Size of the following formula data (RPN token array) (sz)
1	sz	Formula data (RPN token array)

Formula in BIFF3-BIFF8:

Offset	Size	Contents
0	2	Size of the following formula data (sz)
2	sz	Formula data (RPN token array)

The contents of a formula are stored in the Reverse-Polish Notation (RPN). This means, first occur all operands of an operation, followed by the respective operator. The operands and operators are called tokens. For instance the simple term 1+2 consists of 3 tokens. Written in RPN the formula is converted to the token list „1“, „2“, „+“. During parsing such an expression operands are pushed onto a stack. An operator pops the needed number of operands from stack, performs the operation and pushes the result back onto the stack.

Other examples for RPN token arrays:

Formula	Token array	Parsing result
2·4+5	2, 4, „·“, 5, „+“	The „·“ pops 4 and 2 and pushes 8, the „+“ pops 5 and 8 and pushes 40. That is the result.
2+4·5	2, 4, 5, „·“, „+“	The „·“ pops 5 and 4 and pushes 20, the „+“ pops 20 and 2 and pushes 22. That is the result.

A token can be a simple integer or floating point value, a string constant, a cell reference or cell range reference or an operator. A token is stored as follows:

Offset	Size	Contents
0	1	Token identifier
[1]	var.	(optional) Additional data for the token

Example of the formula for the term 1+2:

Offset	Size	Data	Name	Comment
0	2	0007_H	sz	Size of the following formula data
2	1	$1E_H$	ptgInt	} Integer value token
3	2	0001_H		
5	1	$1E_H$	ptgInt	} Integer value token
6	2	0002_H		
8	1	03_H	ptgAdd	Addition operator

In the following token descriptions, only the additional data following the token identifier is described.

3.2 Operators

There are 3 types of operators:

- Unary operators like the minus sign that negates a value. These operators pop the topmost operand from the stack.
- Binary operators like addition or multiplication. These operators pop the two topmost operands from the stack.
- Function operators represent the sheet functions of Excel. They operate on different numbers of topmost operands on the stack. Either the function expects a fixed number of operands (for instance SIN expects one operand) or a variable number of operands given in the function token (for instance SUM is able to process from 0 to 30 operands).

3.3 Reference Classes

Some of the tokens (especially function operators and operand tokens) exist in 3 different versions: reference class token, value class token and array class token. The token class depends on which type of data the involved operator expects. Sometimes only 1 or 2 token classes make sense.

- Reference class token: The reference itself, independent of the cell contents.
- Value class token: A dereferenced value.
- Array class token: A matrix reference to a cell range.

The structure of the 8-bit operand token identifier is described in the following table.

Bit	Mask	Contents
4-0	$1F_H$	Basic token identifier
6-5	60_H	01_2 = Reference class token (token range 20_H-$3F_H$) 10_2 = Value class token (token range 40_H-$5F_H$) 11_2 = Array class token (token range 60_H-$7F_H$)
7	80_H	0_2 (zero)

The class of an operand token is marked in its name: The names of value class tokens contain a trailing „V“ and the names of array class tokens a trailing „A“.

Examples for the different token classes:

- Reference class token: The formula `=ROW(A1)` returns 1, regardless of the content of A1. Cell reference token is ptgRef (24_H).
- Value class token: The formula `=A1+1` returns the value of the cell A1, increased by 1. Cell reference token is ptgRefV (44_H).
- Array class token: The formula `=MDETERM(A1:C3)` returns the determinant of the values inside of the matrix range A1:C3. Area reference token is ptgAreaA (65_H).

3.4 Encoding of Cell References in Tokens

3.4.1 Cell references in BIFF2-BIFF7

In the BIFF versions up to BIFF5, it is possible to use 16384 rows (2^{14}). A cell reference contains the row index as a 16-bit-value (zero-based, 0-16383), the column index as an 8-bit-value (zero-based, 0-255) and two flags. The flags specify whether the row or column index is absolute or relative.

Contents of the row index (16-bit-value), BIFF2-BIFF7:

Bit	Mask	Contents	
13-0	$3FFF_H$	Index to row (0-16383)	
14	4000_H	0 = Absolute column reference	1 = Relative column reference
15	8000_H	0 = Absolute row reference	1 = Relative row reference

Example: The reference B$6 has the absolute row index 5 and the relative column index 1. The value of the encoded row index is 4005_H (row 6, column is relative). The value of the column index is 01_H (column B).

3.4.2 Cell references in BIFF8

From BIFF8 on 65536 (2^{16}) rows are available. Therefore the column index field expands to a 16-bit-value and contains the relative flags.

Contents of the column index (16-bit-value), BIFF8:

Bit	Mask	Contents	
7-0	$00FF_H$	Index to column (0-255)	
14	4000_H	0 = Absolute column reference	1 = Relative column reference
15	8000_H	0 = Absolute row reference	1 = Relative row reference

Example: The reference B$6 has the absolute row index 5 and the relative column index 1. The value of the encoded column index is 4001_H (column B, column is relative). The value of the row index is 0005_H (row 6).

3.5 Token Overview

Following a list of all tokens, separated into several token classes and ordered by token identifier.

3.5.1 Unary operator tokens

Token ID	Token name	Description
12_H	ptgUplus	Unary plus
13_H	ptgUminus	Unary minus
14_H	ptgPercent	Percent sign

3.5.2 Binary operator tokens

Token ID	Token name	Description
03_H	ptgAdd	Addition
04_H	ptgSub	Subtraction
05_H	ptgMul	Multiplication
06_H	ptgDiv	Division
07_H	ptgPower	Exponentiation
08_H	ptgConcat	Concatenation
09_H	ptgLT	Less than
$0A_H$	ptgLE	Less than or equal
$0B_H$	ptgEQ	Equal
$0C_H$	ptgGE	Greater than or equal
$0D_H$	ptgGT	Greater than
$0E_H$	ptgNE	Not equal
$0F_H$	ptgIsect	Cell range intersection
10_H	ptgUnion	Cell range union
11_H	ptgRange	Cell range

3.5.3 Function operator tokens

Token ID	Token name	Description
21_H 41_H 61_H	ptgFunc	Function with fixed number of arguments
22_H 42_H 62_H	ptgFuncVar	Function with variable number of arguments

3.5.4 Constant operand tokens

Token ID	Token name	Description
16_H	ptgMissArg	Missing argument
17_H	ptgStr	String constant
$1C_H$	ptgErr	Error value
$1D_H$	ptgBool	Boolean value
$1E_H$	ptgInt	Integer value
$1F_H$	ptgNum	Floating-point number

3.5.5 Operand tokens

Token ID	Token name	Description
20_H 40_H 60_H	ptgArray	Array constant
23_H 43_H 63_H	ptgName	Internal defined name
24_H 44_H 64_H	ptgRef	2D cell reference
25_H 45_H 65_H	ptgArea	2D area reference
$2A_H$ $4A_H$ $6A_H$	ptgRefErr	Deleted 2D cell reference
$2B_H$ $4B_H$ $6B_H$	ptgAreaErr	Deleted 2D area reference
39_H 59_H 79_H	ptgNameX	External name
$3A_H$ $5A_H$ $7A_H$	ptgRef3d	3D cell reference
$3B_H$ $5B_H$ $7B_H$	ptgArea3d	3D area reference
$3C_H$ $5C_H$ $7C_H$	ptgRefErr3d	Deleted 3D cell reference
$3D_H$ $5D_H$ $7D_H$	ptgAreaErr3d	Deleted 3D area reference
2do: more		

3.6 Unary Operator Tokens

Unary operators perform an operation with the topmost operand from stack. The tokens do not contain any additional data.

3.6.1 ptgUplus (12_H)

Unary plus operator. This operator has no effect on the operand.
Example: `+1` returns 1.

3.6.2 ptgUminus (13_H)

Unary minus operator. Negates the operand.
Example: `-1` returns -1.

3.6.3 ptgPercent (14_H)

Percent sign. Divides the operand by 100.
Example: `1%` returns 0.01.

3.7 Binary Operator Tokens

Binary operators perform an operation with the two topmost operands from stack. The tokens do not contain any additional data.

3.7.1 ptgAdd (03_H)

Addition operator. Adds the operands.
Example: `3+2` returns 5.

3.7.2 ptgSub (04_H)

Subtraction operator. Subtracts the top operand from the second-to-top operand.
Example: `3-2` returns 1.

3.7.3 ptgMul (05_H)

Multiplication operator. Multiplicates the operands.
Example: `3*2` returns 6.

3.7.4 ptgDiv (06_H)

Division operator. Divides the second-to-top operand by the top operand.
Example: `3/2` returns 1.5.

3.7.5 ptgPower (07_H)

Exponentiation operator. Raises the second-to-top operand to the power of the top operand.
Example: `3^2` returns 9.

3.7.6 ptgConcat (08_H)

Concatenation operator. Appends the top operand to the second-to-top operand.
Example: `"ABC"&"DEF"` returns "ABCDEF“.

3.7.7 ptgLT (09_H)

Less than operator. Returns TRUE if the second-to-top operand is less than the top operand.
Example: `3<2` returns FALSE.

3.7.8 ptgLE ($0A_H$)

Less than or equal operator. Returns TRUE if the second-to-top operand is less than or equal to the top operand.
Example: `3<=2` returns FALSE.

3.7.9 ptgEQ ($0B_H$)

Equality operator. Returns TRUE if the operands are equal.
Example: `3=2` returns FALSE.

3.7.10 ptgGE ($0C_H$)

Greater than or equal operator. Returns TRUE if the second-to-top operand is greater than or equal to the top operand.
Example: `3>=2` returns TRUE.

3.7.11 ptgGT ($0D_H$)

Greater than operator. Returns TRUE if the second-to-top operand is greater than the top operand.
Example: `3>2` returns TRUE.

3.7.12 ptgNE ($0E_H$)

Inequality operator. Returns TRUE if the operands are not equal.
Example: `3<>2` returns TRUE.

3.7.13 ptgIsect ($0F_H$)

Intersection operator, represented by the space sign. Returns the intersected range of two ranges.
Example: `A1:B3 B2:C3` returns B2:B3.

3.7.14 ptgUnion (10_H)

Union operator, represented by the comma sign (for instance english Excel) or semicolon (for instance german Excel). Returns the union of two ranges.
Example: `(A1:A2,A2:A3)` will be handled as one parameter (useful for function parameters).

3.7.15 ptgRange (11_H)

Range operator, represented by the colon sign. Returns the rectangular range formed by two ranges. This token occurs for instance by using defined names.
Example: `namedcell:D5`.

3.8 Function Operator Tokens

3.8.1 ptgFunc (21_H), ptgFuncV (41_H), ptgFuncA (61_H)

This token contains the index to a function with fixed number of arguments.
Token ptgFunc, BIFF2-BIFF3:

Offset	Size	Contents
0	1	Index to a sheet function

Token ptgFunc, BIFF4-BIFF8:

Offset	Size	Contents
0	2	Index to a sheet function

3.8.2 ptgFuncVar (21_H), ptgFuncVarV (41_H), ptgFuncVarA (61_H)

This token contains the index to a function with variable number of arguments.
Token ptgFuncVar, BIFF2-BIFF3:

Offset	Size	Contents
0	1	Number of arguments
1	1	Index to a sheet function

Token ptgFuncVar, BIFF4-BIFF8:

Offset	Size	Contents
0	1	Number of arguments
1	2	Index to a sheet function

3.9 Constant Operand Tokens

3.9.1 ptgMissArg (16_H)

A missing argument in a function argument list is stored as a ptgMissArg token. This token does not contain any additional data.

Example: `SUM(1,,3)` – second argument is missing and represented by a ptgMissArg token.

3.9.2 ptgStr (17_H)

This token contains a string constant. The maximum length of the string is 253 characters in BIFF2 (due to the limitation of 255 bytes per formula) and 255 characters in BIFF3-BIFF7.

Token ptgStr, BIFF2-BIFF7:

Offset	Size	Contents
0	var.	Byte string, 8-bit string length (→2.1)

Token ptgStr, BIFF8:

Offset	Size	Contents
0	var.	Unicode string, 16-bit string length, option flags occur always (→2.2)

Example: `"ABC"`.

3.9.3 ptgErr ($1C_H$)

This token contains an error code.

Offset	Size	Contents
0	1	Error code (→2.4)

3.9.4 ptgBool ($1D_H$)

This token contains a boolean value (TRUE or FALSE).

Offset	Size	Contents
0	1	0 = FALSE, 1 = TRUE

3.9.5 ptgInt ($1E_H$)

This token contains an unsigned 16-bit-integer value in the range from 0 to 65535.

Offset	Size	Contents
0	2	Unsigned integer value

3.9.6 ptgNumber ($1F_H$)

This token contains an IEEE floating-point number.

Offset	Size	Contents
0	8	IEEE floating-point number

3.10 Operand Tokens

3.10.1 ptgArray (20_H), ptgArrayV (40_H), ptgArrayA (60_H)

This token contains an array constant. For instance the 2x1 matrix {1;2} is an array constant. The values of the array constant do not follow the token identifier but are stored behind the complete token array.

Token ptgArray, BIFF2-BIFF8:

Offset	Size	Contents
0	7	Not used

The constants of the array are stored row by row behind the formula in a list. The length of this list has not been added to the leading formula size field.

Array constant list, BIFF2-BIFF7:

Offset	Size	Contents
0	1	Number of columns (nc). The value 0 means 256 columns.
1	2	Number of rows (nr)
3	var.	List of nc·nr cached values (→2.5)

Array constant list, BIFF8:

Offset	Size	Contents
0	1	Number of columns decreased by 1 (nc)
1	2	Number of rows decreased by 1 (nr)
3	var.	List of (nc+1)·(nr+1) cached values (→2.5)

3.10.2 ptgName (23_H), ptgNameV (43_H), ptgNameA (63_H)

This token contains the one-based index to a NAME record (→5.25). In BIFF2-BIFF4 this could be the index to an EXTERNNAME record (→5.12) too. From BIFF5 on an external name is represented by the token ptgNameX (→3.10.7).

Token ptgName, BIFF2:

Offset	Size	Contents
0	2	One-based index to NAME record (→5.25) or EXTERNNAME record (→5.12)
2	5	Not used

Token ptgName, BIFF3-BIFF4:

Offset	Size	Contents
0	2	One-based index to NAME record (→5.25) or EXTERNNAME record (→5.12)
2	8	Not used

Token ptgName, BIFF5/BIFF7:

Offset	Size	Contents
0	2	One-based index to NAME record (→5.25)
2	12	Not used

Token ptgName, BIFF8:

Offset	Size	Contents
0	2	One-based index to NAME record (→5.25)
2	2	Not used

3.10.3 ptgRef (24_H), ptgRefV (44_H), ptgRefA (64_H)

This token contains the reference to a cell in the same sheet.
Token ptgRef, BIFF2-BIFF7:

Offset	Size	Contents
0	2	Index to row and relative flags (→3.4.1)
2	1	Index to column

Token ptgRef, BIFF8:

Offset	Size	Contents
0	2	Index to row
2	2	Index to column and relative flags (→3.4.2)

3.10.4 ptgArea (25_H), ptgAreaV (45_H), ptgAreaA (65_H)

This token contains the reference to a rectangular cell range in the same sheet.
Token ptgArea, BIFF2-BIFF7:

Offset	Size	Contents
0	2	Index to first row and relative flags (→3.4.1)
2	2	Index to last row and relative flags (→3.4.1)
4	1	Index to first column
5	1	Index to last column

Token ptgArea, BIFF8:

Offset	Size	Contents
0	2	Index to first row
2	2	Index to last row
4	2	Index to first column and relative flags (→3.4.2)
6	2	Index to last column and relative flags (→3.4.2)

3.10.5 ptgRefErr ($2A_H$), ptgRefErrV ($4A_H$), ptgRefErrA ($6A_H$)

This token contains the last reference to a deleted cell in the same sheet. The structure is equal to the token ptgRef (→3.10.3).

3.10.6 ptgAreaErr ($2B_H$), ptgAreaErrV ($4B_H$), ptgAreaErrA ($6B_H$)

This token contains the last reference to a deleted rectangular cell range in the same sheet. The structure is equal to the token ptgArea (→3.10.4).

3.10.7 ptgNameX (39_H), ptgNameXV (59_H), ptgNameXA (79_H) (BIFF5-BIFF8)

This token contains the index to a NAME or EXTERNNAME record. It occurs by using internal or external names, AddIn functions, DDE links or linked OLE objects. See →4.5.2 for details about references in BIFF5/BIFF7 and →4.5.3 for BIFF8.

Token ptgNameX, BIFF5/BIFF7:

Offset	Size	Contents
0	2	One-based index to EXTERNSHEET record (→5.13). A negative value indicates the own workbook. In this case a NAME record is indexed below. The absolute value indexes to the EXTERNSHEET record that contains the sheet name.
2	8	Not used
10	2	One-based index to NAME record (→5.25) or EXTERNNAME record (→5.12)
12	12	Not used

Token ptgNameX, BIFF8:

Offset	Size	Contents
0	2	Index to REF entry in EXTERNSHEET record (→5.13)
2	2	One-based index to NAME record (→5.25) or EXTERNNAME record (→5.12)
4	2	Not used

3.10.8 ptgRef3d ($3A_H$), ptgRef3dV ($5A_H$), ptgRef3dA ($7A_H$) (BIFF5-BIFF8)

This token contains a 3D reference or an external reference to a cell. See →4.5.2 for details about references in BIFF5/BIFF7 and →4.5.3 for BIFF8.

Token ptgRef3d, BIFF5/BIFF7:

Offset	Size	Contents
0	2	One-based index to EXTERNSHEET record (→5.13). A negative value indicates a 3D reference to the own workbook. The absolute value indexes to the EXTERNSHEET record that contains the first sheet name.
2	8	Not used
10	2	3D reference: Index of first referenced sheet; External reference: Not used
12	2	3D reference: Index of last referenced sheet; External reference: Not used
14	2	Index to row and relative flags (→3.4.1)
16	1	Index to column

Token ptgRef3d, BIFF8:

Offset	Size	Contents
0	2	Index to REF entry in EXTERNSHEET record (→5.13)
2	2	Index to row
4	2	Index to column and relative flags (→3.4.2)

3.10.9 ptgArea3d ($3B_H$), ptgArea3dV ($5B_H$), ptgArea3dA ($7B_H$) (BIFF5-BIFF8)

This token contains a 3D reference or an external reference to a rectangular cell range. See →4.5.2 for details about references in BIFF5/BIFF7 and →4.5.3 for BIFF8.

Token ptgArea3d, BIFF5/BIFF7:

Offset	Size	Contents
0	2	One-based index to EXTERNSHEET record (→5.13). A negative value indicates a 3D reference to the own workbook. The absolute value indexes to the EXTERNSHEET record that contains the first sheet name.
2	8	Not used
10	2	3D reference: Index of first referenced sheet; External reference: Not used
12	2	3D reference: Index of last referenced sheet; External reference: Not used
14	2	Index to first row and relative flags (→3.4.1)
16	2	Index to last row and relative flags (→3.4.1)
18	1	Index to first column
19	1	Index to last column

Token ptgArea3d, BIFF8:

Offset	Size	Contents
0	2	Index to REF entry in EXTERNSHEET record (→5.13)
2	2	Index to first row
4	2	Index to last row
6	2	Index to first column and relative flags (→3.4.2)
8	2	Index to last column and relative flags (→3.4.2)

3.10.10 ptgRefErr3d ($3C_H$), ptgRefErr3dV ($5C_H$), ptgRefErr3dA ($7C_H$) (BIFF5-BIFF8)

This token contains the last 3D reference or external reference to a deleted cell. The structure is equal to the token ptgRef3d (→3.10.8).

3.10.11 ptgAreaErr3d ($3D_H$), ptgAreaErr3dV ($5D_H$), ptgAreaErr3dA ($7D_H$) (BIFF5-BIFF8)

This token contains the last 3D reference or external reference to a deleted rectangular cell range. The structure is equal to the token ptgArea3d (→3.10.9).

4 Worksheet/Workbook Structure

In an Excel file, some complex features are splitted into several records. To keep these features consistent, the position and order of the records is very important. This chapter contains details about the correct combination of the records inside of the stream. The internal structure of the records is described in chapter 5.

4.1 Worksheet Stream (BIFF2-BIFF4)

The whole worksheet file consists of the worksheet stream. All records of the worksheet are enclosed by a leading BOF record (→5.4) and a trailing EOF record (→5.10). The sheet contents section contains all information about the worksheet, for instance sheet dimension, view settings, a font list, a list of defined names and external references, of course the contents and formats of all cells, row heights, column widths, drawing objects, chart objects, etc.

Common structure of a worksheet stream:

BOF	Type = worksheet
	Sheet contents
EOF	

4.2 Workbook Stream (BIFF4)

The whole BIFF4 workbook file consists of the workbook stream. It contains a workbook globals section and a list of the worksheets. The sheets are embedded into the outer pair of BOF/EOF records. The workbook globals section contains common information about the workbook, for instance text encoding, global view settings or a list of all sheet names. Additionally, in each workbook a SHEETSOFFSET record (→5.32) is present. The data of the sheets is stored in worksheet substreams. Each substream is preceded by a SHEETHDR record (→5.31) which contains the name of the sheet and the size of the following substream. The SHEETSOFFSET record mentioned above contains the stream position of the first SHEETHDR record. The substreams have the same structure as described in chapter 4.1. Note: In this context the term „substream“ is only a sequence of records and not a storage sub stream of OLE2 storages.

Common structure of a workbook stream with two sheets, BIFF4:

BOF	Type = workbook globals
	Workbook globals
SHEETSOFFSET	Position of the first SHEETHDR record
	Workbook globals
SHEETHDR	Sheet name = „Sheet1“, Byte length of following BOF/EOF record block
BOF	Type = worksheet
	Sheet contents
EOF	
SHEETHDR	Sheet name = „Sheet2“, Byte length of following BOF/EOF record block
BOF	Type = worksheet
	Sheet contents
EOF	
EOF	

4.3 Workbook Stream (BIFF5-BIFF8)

From BIFF5 on an Excel document is stored as an OLE2 storage. The workbook stream is located in the root directory of the storage. In BIFF5/BIFF7 it is named „Book“, in BIFF8 „Workbook“. The names are case-sensitive. In difference to the BIFF4 workbook stream, the worksheet substreams are appended to the workbook globals section, not embedded. The workbook global section and sheet contents section have similar contents as described for BIFF4 workbooks (→4.2).

Common structure of a workbook stream with two sheets, BIFF5-BIFF8:

BOF	Type = workbook globals
	Workbook globals
EOF	
BOF	Type = worksheet
	Sheet contents
EOF	
BOF	Type = worksheet
	Sheet contents
EOF	

4.4 Shared String Table (BIFF8)

A BIFF8 workbook collects all strings of all text cells in a global list, the shared string table (SST). This table is located in the workbook globals section in the record SST (→5.33). An SST record is followed by an EXTSST record (→5.14) which stores stream positions for a string hash table. Text cells are represented by LABELSST records (→5.22) which contain indexes to the shared string table. For reading Excel files only the SST record and the LABELSST records are important.

Example: A workbook contains anywhere the strings „AAA“, „BBB“ and „CCC“.

BOF	Type = workbook globals
	Workbook globals
SST	String 0 = „AAA“ String 1 = „BBB“ String 2 = „CCC“
EXTSST	
	Workbook globals
EOF	
BOF	Type = worksheet
	Sheet contents
LABELSST	String = 0
LABELSST	String = 2
	Sheet contents
LABELSST	String = 1
LABELSST	String = 0
	Sheet contents
EOF	

4.5 Internal and External References

This chapter describes all types of 3D and external references. In detail, this could be:

- a reference to a cell or a cell range of another sheet in the same workbook (3D reference),
- a reference to a cell or a cell range of a sheet in another workbook (external reference),
- a reference to a global or local defined name (internal name),
- a reference to a defined name in another workbook (external name),
- an AddIn function,
- a DDE link,
- an OLE object link.

For external references and external names a combination of XCT and CRN records occurs which store values of cells of the document. In the case the document cannot be found these values will be used to get the result of an external reference. An XCT record (→5.36) contains the number of following CRN records. A CRN record (→5.7) stores the contents of one cell or a sequence of cells of one row. Fragmentary cell ranges or cell ranges spanning over more than one row are splitted into several CRN records. 3D references do not use these records because the referenced cells are located in the own document.

4.5.1 References in BIFF2-BIFF4

2do

4.5.2 References in BIFF5/BIFF7

The document names and sheet names of references are stored in a list of EXTERNSHEET records. Each worksheet contains an EXTERNSHEET list with documents referenced from this sheet. Formulas in the sheet use indexes to the EXTERNSHEET list.

The XCT and CRN records occur behind the last EXTERNNAME record as far as they exist, otherwise directly behind the respective EXTERNSHEET record.

• External and 3D references

External and 3D references are represented in a formula by the tokens ptgRef3d (→3.10.8) or ptgArea3d (→3.10.9). These tokens contain an index to an EXTERNSHEET record located in the own worksheet and indexes to the first and last referenced sheet.

For 3D references, the tokens contain a negative EXTERNSHEET index, indicating a reference into the own workbook. The absolute value is the one-based index of the EXTERNSHEET record that contains the name of the first sheet. If the referenced sheets do not exist anymore, these tokens contain the sheet indexes $FFFF_H$ (deleted 3D reference).

Each external reference contains the one-based index to an EXTERNSHEET record. The sheet indexes of the tokens are not used.

Example: A document with 7 sheets (named from „Sheet1“ to „Sheet7“) contains the formulas
`=Sheet2!A1`,
`=SUM(Sheet4:Sheet6!A1:B3)`,
`=SUM([example.xls]ExtSheet1!A1:B2)` (contents: A1=1.11, B1=2.22, A2=3.33, B2=4.44),
`=[example.xls]ExtSheet3!A1` (contents: „ABCD“) and
`=Sheet8!A1`.

EXTERNSHEET 1	Name = „Sheet2“
EXTERNSHEET 2	Name = „Sheet4“
EXTERNSHEET 3	Name = „Sheet6“
EXTERNSHEET 4	Name = „[example.xls]ExtSheet1“
XCT	Number of CRN = 2
CRN 0	Cell range = A1:B1, contents = 1.11, 2.22
CRN 1	Cell range = A2:B2, contents = 3.33, 4.44
EXTERNSHEET 5	Name = „[example.xls]ExtSheet3“
XCT	Number of CRN = 1
CRN 0	Cell range = A1, contents = „ABCD“
EXTERNSHEET 6	Name = „Sheet8“

• Internal names

2do

• External names

2do

• AddIn functions

2do

• DDE links, OLE object links

2do

4.5.3 References in BIFF8

The main data of all types of references is stored in a list inside of the workbook globals section. All formulas use only indexes to use specific references. In BIFF8 each referenced document is represented by a SUPBOOK record (→5.35). A SUPBOOK contains the name of the document and the names of the sheets of the document. After the last SUPBOOK occurs only one EXTERNSHEET record (→5.13). It contains a list with indexes to the SUPBOOKs for each used reference anywhere in the document. Formulas use indexes into this EXTERNSHEET list.

For the following examples an external document „example.xls“ is used. It contains 3 sheets named „ExtSheet1“, „ExtSheet2“ and „ExtSheet3“.

Example: A document contains (among other references) the two formulas
`=[example.xls]ExtSheet2!A1` and
`=[example.xls]ExtSheet1!A1`.

	Workbook globals
SUPBOOK 0	Any content
SUPBOOK 1	Document = „example.xls“ Sheet 0 = „ExtSheet1“ Sheet 1 = „ExtSheet2“ Sheet 2 = „ExtSheet3“
SUPBOOK 2	Any content
EXTERNSHEET	REF 0 = any reference REF 1 = {SUPBOOK = 1, sheet range = 1…1} REF 2 = any reference REF 3 = {SUPBOOK = 1, sheet range = 0…0} REF 4 = any reference
	Workbook globals

The first formula uses REF 1 in the EXTERNSHEET record. REF 1 refers to SUPBOOK 1 and sheet range 1…1. This means, the document „example.xls“ is used (document of SUPBOOK 1) and the name of the sheet is „ExtSheet2“ (sheet 1 of SUPBOOK 1). In the same way, the second formula uses REF 3 in the EXTERNSHEET record. All list entries inside of the EXTERNSHEET record are unique. For instance all formulas in the workbook referring to sheet „ExtSheet2“ of the document „example.xls“ use REF 1. All other SUPBOOKs and REFs are placeholders for other references in this example.

The XCT and CRN records occur behind the EXTERNNAME records as far as they exist, otherwise directly behind the respective SUPBOOK record.

• External and 3D references

The SUPBOOK for the own document has a special format: It contains only the number of all sheets and the value 0401_H instead of the sheet names. The sheet range indexes in the EXTERNSHEET record refer to the position of the sheets (zero-based). If a referenced sheet does not exist anymore, the sheet index $FFFF_H$ will occur (deleted 3D reference).

Example: A document with 7 sheets (named from „Sheet1“ to „Sheet7“) contains the formulas
`=Sheet2!A1`,
`=SUM(Sheet4:Sheet6!A1:B3)`,
`=SUM([example.xls]ExtSheet1!A1:B2)` (contents: A1=1.11, B1=2.22, A2=3.33, B2=4.44),
`=[example.xls]ExtSheet3!A1` (contents: „ABCD“) and
`=Sheet8!A1`.

SUPBOOK 0	Number of sheets: 7 0401_H (own workbook)
SUPBOOK 1	Document = „example.xls“ Sheet 0 = „ExtSheet1“ Sheet 1 = „ExtSheet2“ Sheet 2 = „ExtSheet3“
XCT	Number of CRN = 2, sheet = 0 (ExtSheet1)
CRN 0	Cell range = A1:B1, contents = 1.11, 2.22
CRN 1	Cell range = A2:B2, contents = 3.33, 4.44
XCT	Number of CRN = 1, sheet = 2 (ExtSheet3)
CRN 0	Cell range = A1, contents = „ABCD“
EXTERNSHEET	REF 0 = {SUPBOOK = 0, sheet range = 1...1} REF 1 = {SUPBOOK = 0, sheet range = 3...5} REF 2 = {SUPBOOK = 1, sheet range = 0...0} REF 3 = {SUPBOOK = 1, sheet range = 1...1} REF 4 = {SUPBOOK = 0, sheet range = $FFFF_H$...$FFFF_H$}

Inside of the first formula the cell reference is represented by the token ptgRef3d (→3.10.8). The second formula contains the token ptgArea3d (→3.10.9).

• Internal names

All internal names are stored in a list of NAME records (→5.25) that follows the EXTERNSHEET record. There exist two types of internal names: global names which are valid in the whole workbook and local names which are attached to a specific sheet. For instance the local name „MyCell“ of the sheet „Sheet1“ can be used from everywhere in the workbook by entering `=Sheet1!MyCell`. Each NAME record contains the name itself and a one-based sheet index. The index zero indicates a global name. If a SUPBOOK contains local names, a special REF entry will be created in the EXTERNSHEET record. It contains the index to the SUPBOOK and the sheet range $FFFE_H$...$FFFE_H$.

Example for internal names: A document contains
- The global name „GlobalName“,
- The local names „Sheet1!Name“ and „Sheet2!Name“ and
- In Sheet1 the formulas `=GlobalName`, `=Name`, `=Sheet1!Name` and `=Sheet2!Name`.

SUPBOOK 0	Number of sheets: 3 0401_H (own workbook)
EXTERNSHEET	REF 0 = {SUPBOOK = 0, sheet range = 0...0} REF 1 = {SUPBOOK = 0, sheet range = $FFFE_H$...$FFFE_H$}
NAME 1	Name = „GlobalName“, sheet = 0 (Global)
NAME 2	Name = „Name“, sheet = 1 (Sheet1)
NAME 3	Name = „Name“, sheet = 2 (Sheet2)

Inside of the formula a global name or a local name of the own sheet is represented by the token ptgName (→3.10.2) with an one-based index to the NAME record list. The first formula in the example above contains the token ptgNameV with index 1 and the second formula the same token with index 2.

Local names from other sheets are represented by the token ptgNameX (→3.10.7) with an index to the special REF entry of the EXTERNSHEET record and an index to the NAME record list. The third formula contains the token ptgNameX with the value {REF = 1, Name = 2} and the last formula the same token with the value {REF = 1, Name = 3}. Ref 1 refers to SUPBOOK 0 and Name 2 or Name 3 refer to the respective NAME records.

• External names

In Excel formulas can use names located in another workbook. In this case for each name an EXTERNNAME record (→5.12) after the SUPBOOK record occurs. The EXTERNNAME record contains the name itself and the one-based index to the sheet. Again the index zero indicates a global name. If a SUPBOOK contains external names, a special REF entry will be created in the EXTERNSHEET record. It contains the index to the SUPBOOK and the sheet range $FFFE_H$...$FFFE_H$.

Example: A document contains the formulas
`=example.xls!GlobalName` (location: ExtSheet1!B22; contents: 22),
`=[example.xls]ExtSheet3!Name` (location: ExtSheet3!C33; contents: „ABCD“) and
`=[example.xls]ExtSheet1!Name` (location: ExtSheet1!A11; contents: 11).

SUPBOOK 0	Document = „example.xls“ Sheet 0 = „ExtSheet1“ Sheet 1 = „ExtSheet2“ Sheet 2 = „ExtSheet3“
EXTERNNAME 1	Name = „GlobalName“, sheet = 0 (Global)
EXTERNNAME 2	Name = „Name“, sheet = 3 (ExtSheet3)
EXTERNNAME 3	Name = „Name“, sheet = 1 (ExtSheet1)
XCT	Number of CRN = 2, sheet = 0 (ExtSheet1)
CRN 0	Cell range = A11, contents = 11
CRN 1	Cell range = B22, contents = 22
XCT	Number of CRN = 1, sheet = 2 (ExtSheet3)
CRN 0	Cell range = C33, contents = „ABCD“
EXTERNSHEET	REF 1 = {SUPBOOK = 0, sheet range = $FFFE_H$...$FFFE_H$}

Inside of a formula an external name is represented by the token ptgNameX (→3.10.7). It contains the index to the special REF entry inside of the EXTERNSHEET record and the index to a EXTERNNAME record (one-based). The second formula in the example above contains the token ptgNameXV with the value {REF = 0, ExtName = 2}. REF 1 refers to SUPBOOK 0 and ExtName 2 refers to EXTERNNAME 2 (of SUPBOOK 0).

• AddIn functions

AddIn functions are stored similar to external names. If a formula uses an AddIn function, a special SUPBOOK containing only the value $3A01_H$ will occur. Behind of this SUPBOOK the names of all used AddIn functions are listed, each inside of an EXTERNNAME record. A special REF entry with the sheet range $FFFE_H$...$FFFE_H$ will be inserted into the EXTERNSHEET reference list.

Example: A document contains the formulas `=ISODD(1)` and `=ISEVEN(1)`.

SUPBOOK 0	$3A01_H$ (AddIn)
EXTERNNAME 1	Name = „ISODD“
EXTERNNAME 2	Name = „ISEVEN“
EXTERNSHEET	REF 0 = {SUPBOOK = 0, sheet range = $FFFE_H$...$FFFE_H$}

• DDE links, OLE object links

DDE links and OLE object links expect the name of the server application (DDE) or the class name (OLE) and the name of a source document. These items are encoded in a SUPBOOK record. The SUPBOOK is followed by EXTERNNAME records with additional data of the links. An EXTERNNAME record for a DDE links contains the item (data source range) and an EXTERNNAME record for an OLE object link contains the identifier of the object data storage.

Example: A document contains a DDE link to the range „Sheet1.A1:B2“ inside of the Calc document „example.sxc“ and an OLE object link to the bitmap file „example.bmp“.

SUPBOOK 0	Server application = „soffice“ Document = „example.sxc“
EXTERNNAME 1	Type = DDE link Item = „Sheet1.A1:B2“
SUPBOOK 1	Class name = „Package“ Document = „example.bmp“
EXTERNNAME 1	Type = OLE object link Storage = 00012345_H (storage name = „LNK00012345“)
EXTERNSHEET	REF 0 = {SUPBOOK = 0, sheet range = $FFFE_H$...$FFFE_H$} REF 1 = {SUPBOOK = 1, sheet range = $FFFE_H$...$FFFE_H$}

Inside of a formula a DDE link is represented by the token ptgNameX (→3.10.7). An OLE object link contains a ptgNameX token inside of its OBJ record.

4.6 Array Formulas, Shared Formulas

2do

4.7 Multiple Operations (Table Operations)

2do

4.8 AutoFilter

2do

4.9 Scenarios

2do

4.10 Web Queries (BIFF8)

2do

5 Worksheet/Workbook Records

5.1 Overview, Ordered by Record IDs

Record ID	Record name	Occurs in BIFF versions					
		2	3	4	5	7	8
0000_H	DIMENSIONS	x					
0001_H	BLANK	x					
0002_H	INTEGER	x					
0003_H	NUMBER	x					
0004_H	LABEL	x					
0005_H	BOOLERR	x					
0006_H	FORMULA	x			x	x	x
0007_H	STRING	x					
0009_H	BOF	x					
$000A_H$	EOF	x	x	x	x	x	x
0013_H	PASSWORD	x	x	x	x	x	x
0016_H	EXTERNCOUNT	x	x	x	x	x	
0017_H	EXTERNSHEET	x	x	x	x	x	x
0018_H	NAME	x			x	x	x
$001E_H$	FORMAT	x	x				
0023_H	EXTERNNAME	x			x	x	x
0031_H	FONT	x			x	x	x
$003C_H$	CONTINUE	x	x	x	x	x	x
0043_H	XF	x					
0044_H	IXFE	x					
0051_H	DCONREF	x	x	x	x	x	x
0059_H	XCT		x	x	x	x	x
$005A_H$	CRN		x	x	x	x	x
$008E_H$	SHEETSOFFSET			x			
$008F_H$	SHEETHDR			x			
0092_H	PALETTE	x	x	x	x	x	x
$00BD_H$	MULRK				x	x	x
$00BE_H$	MULBLANK				x	x	x
$00E0_H$	XF				x	x	x
$00FC_H$	SST						x
$00FD_H$	LABELSST						x
$00FF_H$	EXTSST						x
$01AE_H$	SUPBOOK						x
$01B8_H$	HLINK						x
0200_H	DIMENSIONS		x	x	x	x	x
0201_H	BLANK		x	x	x	x	x
0203_H	NUMBER		x	x	x	x	x
0204_H	LABEL		x	x	x	x	

Record ID	Record name	Occurs in BIFF versions					
		2	3	4	5	7	8
0205_H	BOOLERR		x	x	x	x	x
0206_H	FORMULA		x				
0207_H	STRING		x	x	x	x	x
0209_H	BOF		x				
0218_H	NAME		x	x			
0223_H	EXTERNNAME		x	x			
0231_H	FONT		x	x			
0243_H	XF		x				
$027E_H$	RK		x	x	x	x	x
0406_H	FORMULA			x			
0409_H	BOF			x			
$041E_H$	FORMAT			x	x	x	x
0443_H	XF			x			
0800_H	SCREENTIP						x
0809_H	BOF				x	x	x
2do: more							

5.2 Overview, Ordered by Record Names

Record ID	Record name	Occurs in BIFF versions					
		2	3	4	5	7	8
0001_H 0201_H	BLANK	x	x	x	x	x	x
$0*09_H$	BOF	x	x	x	x	x	x
0005_H 0205_H	BOOLERR	x	x	x	x	x	x
$003C_H$	CONTINUE	x	x	x	x	x	x
$005A_H$	CRN		x	x	x	x	x
0051_H	DCONREF	x	x	x	x	x	x
0000_H 0200_H	DIMENSIONS	x	x	x	x	x	x
$000A_H$	EOF	x	x	x	x	x	x
0016_H	EXTERNCOUNT	x	x	x	x	x	
0023_H 0223_H	EXTERNNAME	x	x	x	x	x	x
0017_H	EXTERNSHEET	x	x	x	x	x	x
$00FF_H$	EXTSST						x
0031_H 0231_H	FONT	x	x	x	x	x	x
$001E_H$ $041E_H$	FORMAT	x	x	x	x	x	x
$0*06_H$	FORMULA	x	x	x	x	x	x
$01B8_H$	HLINK						x
0002_H	INTEGER	x					
0044_H	IXFE	x					
0004_H 0204_H	LABEL	x	x	x	x	x	
$00FD_H$	LABELSST						x
$00BE_H$	MULBLANK				x	x	x
$00BD_H$	MULRK				x	x	x
0018_H 0218_H	NAME	x	x	x	x	x	x
0003_H 0203_H	NUMBER	x	x	x	x	x	x
0092_H	PALETTE	x	x	x	x	x	x
0013_H	PASSWORD	x	x	x	x	x	x
$027E_H$	RK		x	x	x	x	x
0800_H	SCREENTIP						x
$008F_H$	SHEETHDR			x			
$008E_H$	SHEETSOFFSET			x			
$00FC_H$	SST						x
0007_H 0207_H	STRING	x	x	x	x	x	x
$01AE_H$	SUPBOOK						x
0059_H	XCT		x	x	x	x	x
$0*43_H$ $00E0_H$	XF	x	x	x	x	x	x
2do: more							

5.3 BLANK

BIFF2	BIFF3	BIFF4	BIFF5	BIFF7	BIFF8
0001_H	0201_H	0201_H	0201_H	0201_H	0201_H

This record represents an empty cell. It contains the cell address and formatting information.
Record BLANK, BIFF2:

Offset	Size	Contents
0	2	Index to row
2	2	Index to column
4	3	Cell attributes (→2.9)

Record BLANK, BIFF3-BIFF8:

Offset	Size	Contents
0	2	Index to row
2	2	Index to column
4	2	Index to XF record (→5.37)

5.4 BOF – Begin of File

BIFF2	BIFF3	BIFF4	BIFF5	BIFF7	BIFF8
0009_H	0209_H	0409_H	0809_H	0809_H	0809_H

The BOF record is the first record of a worksheet, the workbook globals section, a chart or a macro sheet.
Record BOF, BIFF2:

Offset	Size	Contents	
0	2	Version	
2	2	Type of the following data:	0010_H = Worksheet 0020_H = Chart 0040_H = Macro sheet

Record BOF, BIFF3:

Offset	Size	Contents	
0	2	Version	
2	2	Type of the following data:	0010_H = Worksheet 0020_H = Chart 0040_H = Macro sheet
4	2	Not used	

Record BOF, BIFF4:

Offset	Size	Contents	
0	2	Version	
2	2	Type of the following data:	0010_H = Worksheet 0020_H = Chart 0040_H = Macro sheet 0100_H = Workbook globals
4	2	Not used	

Record BOF, BIFF5/BIFF7:

Offset	Size	Contents	
0	2	Version	
2	2	Type of the following data:	0005_H = Workbook globals 0006_H = Visual Basic module 0010_H = Worksheet 0020_H = Chart 0040_H = BIFF4 Macro sheet 0100_H = BIFF4 Workbook globals
4	2	Build identifier	
6	2	Build year	

Record BOF, BIFF8:

Offset	Size	Contents	
0	2	Version, contains 0600_H for BIFF8	
2	2	Type of the following data:	0005_H = Workbook globals 0006_H = Visual Basic module 0010_H = Worksheet 0020_H = Chart 0040_H = BIFF4 Macro sheet 0100_H = BIFF4 Workbook globals
4	2	Build identifier	
6	2	Build year	
8	4	File history flags	
12	4	Lowest Excel version that can read all records in this file	

5.5 BOOLERR

BIFF2	BIFF3	BIFF4	BIFF5	BIFF7	BIFF8
0005_H	0205_H	0205_H	0205_H	0205_H	0205_H

This record represents a boolean or error value cell.

Record BOOLERR, BIFF2:

Offset	Size	Contents
0	2	Index to row
2	2	Index to column
4	3	Cell attributes (→2.9)
7	1	Boolean or error value, depending on the following byte
8	1	0 = Boolean value; 1 = Error code

Record BOOLERR, BIFF3-BIFF8:

Offset	Size	Contents
0	2	Index to row
2	2	Index to column
4	2	Index to XF record (→5.37)
6	1	Boolean or error value, depending on the following byte
7	1	0 = Boolean value; 1 = Error code

If the value field is a boolean value, it will contain 0 for FALSE and 1 for TRUE. See →2.4 for a list of error codes.

5.6 CONTINUE

BIFF2	BIFF3	BIFF4	BIFF5	BIFF7	BIFF8
$003C_H$	$003C_H$	$003C_H$	$003C_H$	$003C_H$	$003C_H$

Everytime the content of a record exceeds the given limits (see table), the record must be splitted. Several CONTINUE records containing the additional data are added after the parent record.

BIFF version	Maximum data size of a record
BIFF2-BIFF7	2080 bytes (2084 bytes including record header)
BIFF8	8224 bytes (8228 bytes including record header)

Record CONTINUE, BIFF2-BIFF8:

Offset	Size	Contents
0	var.	data continuation of the previous record

Unicode strings are splitted in a special way. At the beginning of each CONTINUE record the option flags byte is repeated. Only the character size flag will be set in this flags byte, the Rich-Text flag and the Far-East flag are set to zero.

Attention: In each CONTINUE record it is possible that the character size changes from 8-bit-characters to 16-bit-characters and vice versa. Never an Unicode string is splitted between character count field and option flags field or between option flags field and first character.

Example: The remaining size of a record may be 10 bytes (it has 8214 bytes of data). Now the string „ABCDEFGHØI“ has to be stored in this record. „Ø“ may be a special character with the character code 1234_H. Note: The records are shown with their headers to make the example clearer.

Offset	Size	Contents	Description
0	2		Any record identifier
2	2	2020_H (8224)	Record data size
4	8214		Any data
8218	2	$000A_H$ (10)	Unicode string character count
8220	1	00_H	Unicode string option flags (8-bit-characters)
8221	7	41_H 42_H ... 47_H	8-bit-character array „ABCDEFG“
8228	2	$003C_H$	Record identifier CONTINUE
8230	2	0007_H (7)	Record data size
8232	1	01_H	Unicode string option flags (16-bit-characters)
8233	2	0048_H	16-bit-character „H“
8235	2	1234_H	16-bit-character „Ø“
8237	2	0049_H	16-bit-character „I“

5.7 CRN

BIFF2	BIFF3	BIFF4	BIFF5	BIFF7	BIFF8
---	$005A_H$	$005A_H$	$005A_H$	$005A_H$	$005A_H$

This record stores the contents of an external cell or cell range. An external cell range has one row only. If a cell range spans over more than one row, several CRN records will be created. See →4.5 for details about external references.

Record CRN, BIFF3-BIFF8:

Offset	Size	Contents
0	1	Index to last column inside of the referenced sheet (lc)
1	1	Index to first column inside of the referenced sheet (fc)
2	2	Index to row inside of the referenced sheet
4	var.	List of lc-fc+1 cached values (→2.5)

5.8 DCONREF – Data Consolidation Reference

BIFF2	BIFF3	BIFF4	BIFF5	BIFF7	BIFF8
0051_H	0051_H	0051_H	0051_H	0051_H	0051_H

2do

5.9 DIMENSIONS

BIFF2	BIFF3	BIFF4	BIFF5	BIFF7	BIFF8
0000_H	0200_H	0200_H	0200_H	0200_H	0200_H

2do

5.10 EOF – End of File

BIFF2	BIFF3	BIFF4	BIFF5	BIFF7	BIFF8
$000A_H$	$000A_H$	$000A_H$	$000A_H$	$000A_H$	$000A_H$

This record has no content. It indicates the end of a record block with leading BOF record (→5.4). This could be the end of the workbook globals, a worksheet, a chart, etc.

5.11 EXTERNCOUNT

BIFF2	BIFF3	BIFF4	BIFF5	BIFF7	BIFF8
0016_H	0016_H	0016_H	0016_H	0016_H	---

This record contains the number of following EXTERNSHEET records. In BIFF8 this record is omitted because there occurs only one EXTERNSHEET record. See →4.5.1 for details about external references in BIFF2-BIFF4 and →4.5.2 for BIFF5/BIFF7.

Record EXTERNCOUNT, BIFF2-BIFF7:

Offset	Size	Contents
0	2	Number of following EXTERNSHEET records (→5.13)

5.12 EXTERNNAME

BIFF2	BIFF3	BIFF4	BIFF5	BIFF8
0023_H	0223_H	0223_H	0023_H	0023_H

This record contains the name of an external defined name, the name of an AddIn function, a DDE link item or an OLE object storage name (BIFF8).

• EXTERNNAME in BIFF2-BIFF7

The meaning of the name is dependent on the leading EXTERNSHEET record (→5.13). See →4.5.1 for details about external references in BIFF2-BIFF4 and →4.5.2 for BIFF5/BIFF7.

Record EXTERNNAME, BIFF2-BIFF7:

Offset	Size	Contents
0	var.	External name (byte string, 8-bit string length, →2.1)

If the record contains an item of a DDE link, a list with cached values will be appended to the string. These values are used as results for the DDE link. They are saved row by row for a DDE link that spans over several cells. Note: Only the results of the DDE link (the contents of the referenced cells) are stored, not the results of the complete formulas.

Record EXTERNNAME for DDE items, BIFF2-BIFF7:

Offset	Size	Contents
0	var.	DDE item (byte string, 8-bit string length, →2.1)
var.	1	Number of columns (nc). The value 0 means 256 columns.
var.	2	Number of rows (nr)
var.	var.	List of nc·nr cached values (→2.5)

• EXTERNNAME in BIFF8

In BIFF8 the record contains option flags which describe the type of the external name. So, this record must follow the correct SUPBOOK record (→5.35) and must contain the correct flags. See →4.5.3 for details about external references in BIFF8.

Record EXTERNNAME for external names and AddIn functions, BIFF8:

Offset	Size	Contents
0	2	Option flags (see below)
2	2	One-based sheet index. The value 0 means all sheets or AddIn function.
4	2	Not used
6	var.	External name or AddIn function name (Unicode string, 8-bit string length, →2.2)
var.	var.	For external names only: formula data (RPN token array, →3)

Record EXTERNNAME for DDE links, BIFF8:

Offset	Size	Contents
0	2	Option flags (see below)
2	4	Not used
6	var.	DDE item (Unicode string, 8-bit string length, →2.2)
var.	1	Number of columns decreased by 1 (nc)
var.	2	Number of rows decreased by 1 (nr)
var.	var.	List of (nc+1)·(nr+1) cached values (→2.5)

Record EXTERNNAME for OLE object links, BIFF8:

Offset	Size	Contents
0	2	Option flags (see below)
2	4	Storage identifier
6	3	01_H 00_H 27_H

Option flags:

Bit	Mask	Contents	
0	0001_H	0 = No BuiltIn name	1 = BuiltIn name
1	0002_H	0 = Manual DDE/OLE link	1 = Automatic DDE/OLE link
4	0010_H	0 = External name or DDE link	1 = OLE object link
14-5	$7FE0_H$	For DDE links only: clipboard format of last successful update	

5.13 EXTERNSHEET

BIFF2	BIFF3	BIFF4	BIFF5	BIFF7	BIFF8
0017_H	0017_H	0017_H	0017_H	0017_H	0017_H

• EXTERNSHEET in BIFF2-BIFF7

In the file format versions up to BIFF7 this record stores the name of an external document and a sheet name inside of this document. See →4.5.1 for details about external references in BIFF2-BIFF4 and →4.5.2 for BIFF5/BIFF7.

Record EXTERNSHEET, BIFF2-BIFF7:

Offset	Size	Contents
0	var.	Encoded document and sheet name (→2.6). Byte string, 8-bit string length (→2.1).

Attention: The string length field is decreased by 1, if the EXTERNSHEET stores a reference to one of the own sheets (first character is 03_H). Example: The formula `=Sheet2!A1` contains a reference to an EXTERNSHEET record with the string „<03_H>Sheet2“. The string consists of 7 characters but the string length field contains the value 6.

If a formula uses an AddIn function, a special EXTERNSHEET record will occur, followed by an EXTERNNAME record with the name of the function.

Record EXTERNSHEET for AddIn functions, BIFF2-BIFF7:

Offset	Size	Contents
0	2	3401_H (01_H 34_H = the byte string „#“)

• EXTERNSHEET in BIFF8

In BIFF8 the record stores a list with indexes to SUPBOOK records (list of REF structures). See →4.5.3 for details about external references in BIFF8.

Record EXTERNSHEET, BIFF8:

Offset	Size	Contents
0	2	Number of following REF structures (nm)
2	6·nm	List of nm REF structures. Each REF contains the following data:

Offset	Size	Contents
0	2	Index to SUPBOOK record
2	2	Index to first SUPBOOK sheet
4	2	Index to last SUPBOOK sheet

5.14 EXTSST – Extended SST

BIFF2	BIFF3	BIFF4	BIFF5	BIFF7	BIFF8
---	---	---	---	---	$00FF_H$

This record occurs in conjunction with the SST record (→5.33). It contains a hash table with stream offsets to the SST record to optimize string search operations. Excel does not shorten this record if strings are deleted from the shared string table, so the last part might contain invalid data. The stream indexes in this record divide the SST into hash buckets containing a constant number of strings. See →4.4 for more information about shared string tables.

Record EXTSST, BIFF8:

Offset	Size	Contents		
0	2	Number of strings in a hash bucket, this number is ⩾8		
2	var.	List of OFFSET structures. Each OFFSET contains the following data:		
		Offset	**Size**	**Contents**
		0	4	Absolute stream position of first string of this bucket
		4	2	Position of first string of this bucket inside of current record, including record header. This counter restarts at zero inside of CONTINUE records.
		6	2	Not used

5.15 FONT

BIFF2	BIFF3	BIFF4	BIFF5	BIFF7	BIFF8
0031_H	0231_H	0231_H	0031_H	0031_H	0031_H

This record contains information about an used font, including character formatting.

Some of the elements occur unchanged in every BIFF version. These elements are described in the following tables using a specific name for each element. In the description of the record structure the names are used to reference to these tables.

5.15.1 FONT substructures

• FONT_SCRIPT – Subscript or superscript (2 bytes), BIFF5-BIFF8

Value	Contents
0000_H	None
0001_H	Superscript
0002_H	Subscript

• FONT_UNDERLINE – Underline type (1 byte), BIFF5-BIFF8

Value	Contents
00_H	None
01_H	Single
02_H	Double
03_H	Single accounting
04_H	Double accounting

5.15.2 FONT record contents

Record FONT, BIFF2:

Offset	Size	Contents		
0	2	Height of the font (in 1/20 of a point)		
2	2	Option flags:		
		Bit	**Mask**	**Contents**
		0	0001_H	1 = Characters are bold
		1	0002_H	1 = Characters are italic
		2	0004_H	1 = Characters are underlined
		3	0008_H	1 = Characters are struck out
4	var.	Font name (byte string, 8-bit string length, →2.1)		

Record FONT, BIFF3-BIFF4:

Offset	Size	Contents		
0	2	Height of the font (in 1/20 of a point)		
2	2	Option flags:		
		Bit	**Mask**	**Contents**
		0	0001_H	1 = Characters are bold
		1	0002_H	1 = Characters are italic
		2	0004_H	1 = Characters are underlined
		3	0008_H	1 = Characters are struck out
4	2	Index into PALETTE record (→5.27)		
6	var.	Font name (byte string, 8-bit string length, →2.1)		

Record FONT, BIFF5-BIFF8:

Offset	Size	Contents		
0	2	Height of the font (in 1/20 of a point)		
2	2	Option flags:		
		Bit	**Mask**	**Contents**
		1	0002_H	1 = Characters are italic
		3	0008_H	1 = Characters are struck out
4	2	Index into PALETTE record (→5.27)		
6	2	Boldness (100-1000). Standard values are 0190_H (400) for normal text and $02BC_H$ (700) for bold text.		
8	2	FONT_SCRIPT – Subscript or superscript (see above)		
10	1	FONT_UNDERLINE – Underline type (see above)		
11	1	Font family...		
12	1	Character set...		
13	1	Not used		
14	var.	Font name:	BIFF5/BIFF7: Byte string, 8-bit string length (→2.1) BIFF8: Unicode string, 8-bit string length (→2.2)	

5.16 FORMAT

BIFF2	BIFF3	BIFF4	BIFF5	BIFF7	BIFF8
$001E_H$	$001E_H$	$041E_H$	$041E_H$	$041E_H$	$041E_H$

2do

5.17 FORMULA

BIFF2	BIFF3	BIFF4	BIFF5	BIFF7	BIFF8
0006_H	0206_H	0406_H	0006_H	0006_H	0006_H

This record contains the token array and the result of a formula cell.

• Record contents

Record FORMULA, BIFF2:

Offset	Size	Contents
0	2	Index to row
2	2	Index to column
4	3	Cell attributes (→2.9)
7	8	Result of the formula (IEEE floating-point value)
15	1	0 = Do not recalculate, 1 = Recalculate always
16	var.	Formula data (RPN token array, →3)

Record FORMULA, BIFF3-BIFF4:

Offset	Size	Contents		
0	2	Index to row		
2	2	Index to column		
4	2	Index to XF record (→5.37)		
6	8	Result of the formula. See below for details.		
14	2	Option flags:		
		Bit	**Mask**	**Contents**
		0	0001_H	1 = Recalculate always
		1	0002_H	1 = Calculate on open
16	var.	Formula data (RPN token array, →3)		

Record FORMULA, BIFF5-BIFF8:

Offset	Size	Contents		
0	2	Index to row		
2	2	Index to column		
4	2	Index to XF record (→5.37)		
6	8	Result of the formula. See below for details.		
14	2	Option flags:		
		Bit	**Mask**	**Contents**
		0	0001_H	1 = Recalculate always
		1	0002_H	1 = Calculate on open
		3	0008_H	1 = Part of a shared formula
16	4	Not used		
20	var.	Formula data (RPN token array, →3)		

• Result of the formula

Dependent on the type of value the formula returns, the result field has the following format:
Result is a numeric value:

Offset	Size	Contents
0	8	IEEE floating-point value

Result is a string (the string itself follows in a STRING record, →5.34):

Offset	Size	Contents
0	1	00_H (identifier for a string value)
1	5	Not used
6	2	$FFFF_H$

Result is a boolean value:

Offset	Size	Contents
0	1	01_H (identifier for a boolean value)
1	1	Not used
2	1	0 = FALSE, 1 = TRUE
3	3	Not used
6	2	$FFFF_H$

Result is an error value:

Offset	Size	Contents
0	1	02_H (identifier for an error value)
1	1	Not used
2	1	Error code (→2.4)
3	3	Not used
6	2	$FFFF_H$

5.18 HLINK – Hyperlink

BIFF2	BIFF3	BIFF4	BIFF5	BIFF7	BIFF8
---	---	---	---	---	$01B8_H$

In Excel, every cell can contain only one hyperlink. Therefore, the HLINK record refers to one cell address or a cell range where all cells contain the same hyperlink. Every hyperlink can contain a text mark and a description that is shown in the sheet instead of the real link. Text marks are appended behind a link, separated by the „#“ sign. Examples for text marks are www.xyz.org#table1 or C:\example.xls#Sheet1!A1.

Inside of this record strings are stored in several formats. Sometimes occurs the character count, otherwise the character array size (in 16-bit-character arrays the character count is half of the array size). Furthermore some strings are zero-terminated, others not. They are stored either as 16-bit-character arrays or as 8-bit-character arrays, independent of the characters.

5.18.1 Common record contents

Each HLINK record starts with the same data items and continues with special data related to the current type of hyperlink.

Record HLINK, BIFF8:

Offset	Size	Contents
0	2	Index to first row
2	2	Index to last row
4	2	Index to first column
6	2	Index to last column
8	20	Unknown byte sequence: $D0_H$ $C9_H$ EA_H 79_H $F9_H$ BA_H CE_H 11_H $8C_H$ 82_H 00_H AA_H 00_H $4B_H$ $A9_H$ $0B_H$ 02_H 00_H 00_H 00_H
28	4	Option flags (see below)
[32]	4	(optional, see option flags) Character count of description text, including trailing zero word (dl)
[36]	2·dl	(optional, see option flags) Character array of description text, no Unicode string header, always 16-bit-characters, zero-terminated
		Special data (➜5.18.2 and following)
[var.]	4	(optional, see option flags) Character count of the text mark, including trailing zero word (tl)
[var.]	2·tl	(optional, see option flags) Character array of the text mark without „#“ sign, no Unicode string header, always 16-bit-characters, zero-terminated

The special data parts in the following are described with relative offsets (starting again by zero). The real offset inside of the record data (without header) is either 32 (without description) or 36+2·dl (with description).

• Option flags

The option flags specify the following content of the record.

Bit	Mask	Contents	
0	00000001_H	0 = No link extant	1 = File link or URL
1	00000002_H	0 = Relative file path	1 = Absolute path or URL
2 and 4	00000014_H	0 = No description	1 (both bits) = Description
3	00000008_H	0 = No text mark	1 = Text mark
8	00000100_H	0 = File link or URL	1 = Network path

5.18.2 Hyperlink to a common URL

These data fields occur for links which are not local files or files in the local network. The lower 9 bits of the option flags field must be $0.000x.xx11_2$ (x means optional, depending on hyperlink content). The byte sequence should be used to distinguish an URL from a file link.

Offset	Size	Contents
0	16	Unknown byte sequence, used as URL identifier: $E0_H$ $C9_H$ EA_H 79_H $F9_H$ BA_H CE_H 11_H $8C_H$ 82_H 00_H AA_H 00_H $4B_H$ $A9_H$ $0B_H$
16	4	Size of character array of the URL, including trailing zero word (us). There are us/2-1 characters in the following string.
20	us	Character array of the URL, no Unicode string header, always 16-bit-characters, zero-terminated

5.18.3 Hyperlink to a local file

These data fields are for links to files on local drives. The path of the file can be complete with drive letter (absolute) or relative to the location of the workbook. The lower 9 bits of the option flags field must be $0.000x.xxx1_2$. The byte sequence should be used to distinguish an URL from a file link.

Offset	Size	Contents
0	16	Unknown byte sequence, used as file link identifier: 03_H 03_H 00_H 00_H 00_H 00_H 00_H 00_H $C0_H$ 00_H 00_H 00_H 00_H 00_H 00_H 46_H
16	2	Directory up-level count. Each leading „..\“ in the file link is deleted and inceases this counter.
18	4	Character count of the shortened file path and name, including trailing zero byte (sl)
22	sl	Character array of the shortened file path and name in 8.3-DOS-format. This field can be filled with a long file name too. No Unicode string header, always 8-bit-characters, zero-terminated.
22+sl	24	Unknown byte sequence: FF_H FF_H AD_H DE_H 00_H 00_H 00_H 00_H 00_H 00_H 00_H 00_H 00_H 00_H 00_H 00_H 00_H 00_H 00_H 00_H 00_H 00_H 00_H 00_H
46+sl	4	Size of the following file link field including string length field and additional data field (sz). If sz is zero, nothing will follow (except a text mark).
[50+sl]	4	(optional) Size of character array of the extended file path and name (xl). There are xl/2 characters in the following string.
[54+sl]	2	(optional) Unknown byte sequence: 03_H 00_H
[56+sl]	xl	(optional) Character array of the extended file path and name (xl), no Unicode string header, always 16-bit-characters, not zero-terminated

5.18.4 Hyperlink to a file located in the local network

These data fields are for links to files located in the local network. The lower 9 bits of the option flags field must be $1.000x.xx11_2$.

Offset	Size	Contents
0	4	Character count of the network path and file name, including trailing zero word (fl)
4	2·fl	Character array of the network path and file name, no Unicode string header, always 16-bit-characters, zero-terminated.

5.18.5 Hyperlink to a place in the current workbook

In this case only the text mark field is present (optional with description). Example: The URL „#Sheet2! B1:C2“ refers to the given range in the current workbook. The lower 9 bits of the option flags field must be $0.000x.1x00_2$.

5.19 INTEGER

BIFF2	BIFF3	BIFF4	BIFF5	BIFF7	BIFF8
0002_H	---	---	---	---	---

This record represents a cell that contains an unsigned 16-bit-integer value. If a value cannot be stored as a 16-bit-integer, a NUMBER record (→5.26) must be written. From BIFF3 on this record is replaced by the RK record (→5.29).

Record INTEGER, BIFF2:

Offset	Size	Contents
0	2	Index to row
2	2	Index to column
4	3	Cell attributes (→2.9)
7	2	Unsigned 16-bit-integer value

5.20 IXFE – Index to XF

BIFF2	BIFF3	BIFF4	BIFF5	BIFF7	BIFF8
0044_H	---	---	---	---	---

This record occurs in front of every cell record (for instance BLANK, INTEGER, NUMBER, LABEL, FORMULA) that references to an XF record (→5.37) with an index greater than 62. The XF index field of the cell record consists only of 6 bits. The maximum value 63 is used to indicate a preceding IXFE record containing the real XF index. See →2.9 for more details.

Record IXFE, BIFF2:

Offset	Size	Contents
0	2	Index to XF record (→5.37)

5.21 LABEL

BIFF2	BIFF3	BIFF4	BIFF5	BIFF7	BIFF8
0004_H	0204_H	0204_H	0204_H	0204_H	---

This record represents a cell that contains a string. In BIFF8 it is replaced by the LABELSST record (→5.22).

Record LABEL, BIFF2:

Offset	Size	Contents
0	2	Index to row
2	2	Index to column
4	3	Cell attributes (→2.9)
7	var.	Byte string, 8-bit string length (→2.1)

Record LABEL, BIFF3-BIFF7:

Offset	Size	Contents
0	2	Index to row
2	2	Index to column
4	2	Index to XF record (→5.37)
6	var.	Byte string, 16-bit string length (→2.1)

5.22 LABELSST

BIFF2	BIFF3	BIFF4	BIFF5	BIFF7	BIFF8
---	---	---	---	---	$00FD_H$

This record represents a cell that contains a string. It replaces the LABEL record (→5.21) used in BIFF2-BIFF7. See →4.4 for more information about shared string tables.

Record LABELSST, BIFF8:

Offset	Size	Contents
0	2	Index to row
2	2	Index to column
4	2	Index to XF record (→5.37)
6	4	Index into SST record (→5.33)

5.23 MULBLANK – Multiple BLANK

BIFF2	BIFF3	BIFF4	BIFF5	BIFF7	BIFF8
---	---	---	$00BE_H$	$00BE_H$	$00BE_H$

This record represents a cell range of empty cells. All cells are located in the same row.

Record MULBLANK, BIFF5-BIFF8:

Offset	Size	Contents
0	2	Index to row
2	2	Index to first column (fc)
4	2·(lc-fc+1)	Array of lc-fc+1 16-bit-indexes to XF records (→5.37)
var.	2	Index to last column (lc)

5.24 MULRK – Multiple RK

BIFF2	BIFF3	BIFF4	BIFF5	BIFF7	BIFF8
---	---	---	$00BD_H$	$00BD_H$	$00BD_H$

This record represents a cell range containing RK value cells. All cells are located in the same row.

Record MULRK, BIFF5-BIFF8:

Offset	Size	Contents
0	2	Index to row
2	2	Index to first column (fc)
4	6·(lc-fc+1)	Array of lc-fc+1 XF/RK structures. Each XF/RK contains:

Offset	Size	Contents
0	2	Index to XF record (→5.37)
2	4	RK value (→2.3)

var.	2	Index to last column (lc)

5.25 NAME

BIFF2	BIFF3	BIFF4	BIFF5	BIFF7	BIFF8
0018_H	0218_H	0218_H	0018_H	0018_H	0018_H

This record contains the name and the token array of an internal defined name.
Record NAME, BIFF2:

Offset	Size	Contents		
0	1	Option flags:		
		Bit	**Mask**	**Contents**
		1	02_H	1 = Function macro or command macro
		2	04_H	1 = Complex function (array formula or user defined)
1	1	If name is function macro or command macro (see option flags above): 01_H = Function macro, 02_H = Command macro		
2	1	Keyboard shortcut		
3	1	Length of the name (character count) (ln)		
4	1	Size of the formula data (RPN token array) (sz)		
5	ln	Character array of the name		
5+ln	sz	Formula data (RPN token array without size field, →3)		
5+ln+sz	1	Duplicate of the formula data size field (sz)		

Record NAME, BIFF3-BIFF4:

Offset	Size	Contents		
0	2	Option flags:		
		Bit	**Mask**	**Contents**
		0	0001_H	1 = Name is hidden
		1	0002_H	1 = Name is a function
		2	0004_H	1 = Name is a command
		3	0008_H	1 = Function macro or command macro
		4	0010_H	1 = Complex function (array formula or user defined)
		5	0020_H	1 = Built-in name (see table below)
		11-6	$0FC0_H$	BIFF3: Not used; BIFF4: Index to function group
2	1	Keyboard shortcut		
3	1	Length of the name (character count) (ln)		
4	2	Size of the formula data (RPN token array) (sz)		
6	ln	Character array of the name		
6+ln	sz	Formula data (RPN token array without size field, →3)		

Record NAME, BIFF5/BIFF7:

Offset	Size	Contents
0	2	Option flags:

Bit	Mask	Contents
11-0	$0FFF_H$	Equal to BIFF4 (see table above)
12	1000_H	1 = Name contains binary data

Offset	Size	Contents
2	1	Keyboard shortcut
3	1	Length of the name (character count) (ln)
4	2	Size of the formula data (RPN token array) (sz)
6	2	Unused
8	2	0 = Global name, otherwise index to sheet (one-based)
10	1	Length of menu text (character count) (lm)
11	1	Length of description text (character count) (ld)
12	1	Length of help topic text (character count) (lh)
13	1	Length of status bar text (character count) (ls)
14	ln	Character array of the name
14+ln	sz	Formula data (RPN token array without size field, →3)
14+ln+sz	lm	Character array of menu text
var.	ld	Character array of description text
var.	lh	Character array of help topic text
var.	ls	Character array of status bar text

Record NAME, BIFF8:

Offset	Size	Contents
0	2	Option flags: Equal to BIFF5/BIFF7 (see table above)
2	1	Keyboard shortcut
3	1	Length of the name (character count)
4	2	Size of the formula data (RPN token array) (sz)
6	2	Unused
8	2	0 = Global name, otherwise index to sheet (one-based)
10	1	Length of menu text (character count)
11	1	Length of description text (character count)
12	1	Length of help topic text (character count)
13	1	Length of status bar text (character count)
14	var.	Name (Unicode string without length field, →2.2)
var.	sz	Formula data (RPN token array without size field, →3)
var.	var.	Menu text (Unicode string without length field, →2.2)
var.	var.	Description text (Unicode string without length field, →2.2)
var.	var.	Help topic text (Unicode string without length field, →2.2)
var.	var.	Status bar text (Unicode string without length field, →2.2)

- **Built-in names**

From BIFF3 on only an index to a built-in names is stored. If bit 5 of the option flags field is set, the name string contains only one character with this index.

Built-in index	Built-In name
00_H	Consolidate_Area
01_H	Auto_Open
02_H	Auto_Close
03_H	Extract
04_H	Database
05_H	Criteria
06_H	Print_Area
07_H	Pint_Titles
08_H	Recorder
09_H	Data_Form
$0A_H$	Auto_Activate
$0B_H$	Auto_Deactivate
$0C_H$	Sheet_Title
$0D_H$	BIFF3-BIFF4: Not used; BIFF5-BIFF8: Autofilter

5.26 NUMBER

BIFF2	BIFF3	BIFF4	BIFF5	BIFF7	BIFF8
0003_H	0203_H	0203_H	0203_H	0203_H	0203_H

This record represents a cell that contains a floating-point value.

Record NUMBER, BIFF2:

Offset	Size	Contents
0	2	Index to row
2	2	Index to column
4	3	Cell attributes (→2.9)
7	8	IEEE floating-point value

Record NUMBER, BIFF3-BIFF8:

Offset	Size	Contents
0	2	Index to row
2	2	Index to column
4	2	Index to XF record (→5.37)
6	8	IEEE floating-point value

5.27 PALETTE

BIFF2	BIFF3	BIFF4	BIFF5	BIFF7	BIFF8
---	0092_H	0092_H	0092_H	0092_H	0092_H

This record contains the definition of all colors available for cell and object formatting.
Record PALETTE, BIFF3-BIFF8:

Offset	Size	Contents
0	2	Number of following colors (nm)
2	4·nm	List of nm colors. Each color contains:

Offset	Size	Contents
0	1	Red component of the color
1	1	Green component of the color
2	1	Blue component of the color
3	1	Not used

5.28 PASSWORD

BIFF2	BIFF3	BIFF4	BIFF5	BIFF7	BIFF8
0013_H	0013_H	0013_H	0013_H	0013_H	0013_H

This record stores a 16-bit hash value for a sheet or workbook protection password.

Offset	Size	Contents
0	2	16-bit hash value of the password

This is the algorithm to create the hash value from a given password:

- The ASCII values of all characters are rotated left with a number of digits depending on the character position (first character is rotated left 1 bit, second character 2 bits, and so on). There is a space of 15 bits available for rotation (bit 15 jumps to bit 0, bit 16 jumps to bit 1 and so on).
- All rotated characters are combined using XOR operation.
- The number of characters is added using XOR operation.
- The constant $CE4B_H$ is added using XOR operation.

Example: The password is „abcdefghij“ (10 characters).

Character	ASCII	Shifted	Rotated
a	61_H	$000000C2_H$	$00C2_H$
b	62_H	00000188_H	0188_H
c	63_H	00000318_H	0318_H
d	64_H	00000640_H	0640_H
e	65_H	$00000CA0_H$	$0CA0_H$
f	66_H	00001980_H	1980_H
g	67_H	00003380_H	3380_H
h	68_H	00006800_H	6800_H
i	69_H	$0000D200_H$	5201_H
j	$6A_H$	$0001A800_H$	2803_H

All the rotated values and the number of characters $000A_H$ and the constant $CE4B_H$ result in the hash value $FEF1_H$.

5.29 RK

BIFF2	BIFF3	BIFF4	BIFF5	BIFF7	BIFF8
---	$027E_H$	$027E_H$	$027E_H$	$027E_H$	$027E_H$

This record represents a cell that contains an RK value (encoded integer or floating-point value). If a floating-point value cannot be encoded to an RK value, a NUMBER record (→5.26) must be written. This record replaces the record INTEGER (→5.19) written in BIFF2.

Record RK, BIFF3-BIFF8:

Offset	Size	Contents
0	2	Index to row
2	2	Index to column
4	2	Index to XF record (→5.37)
6	4	RK value (→2.3)

5.30 SCREENTIP

BIFF2	BIFF3	BIFF4	BIFF5	BIFF7	BIFF8
---	---	---	---	---	0800_H

This record contains the cell range and text for a screen tip. It occurs in conjunction with the HLINK record for hyperlinks (→5.18).

Record SCREENTIP, BIFF8:

Offset	Size	Contents
0	2	0800_H (repeated record ID)
2	2	Index to first row
4	2	Index to last row
6	2	Index to first column
8	2	Index to last column
10	var.	Character array of the screen tip, no Unicode string header, always 16-bit-characters, zero-terminated

5.31 SHEETHDR

BIFF2	BIFF3	BIFF4	BIFF5	BIFF7	BIFF8
---	---	$008F_H$	---	---	---

This record occurs only in BIFF4 workbook files. It precedes a substream for a sheet (delimited by a BOF and a EOF record) and contains the byte length of the substream and the sheet name. Adding the substream length to the stream position of the following BOF record gives the position of the next SHEETHDR record. See →4.2 for details about the BIFF4 workbook stream.

Record SHEETHDR, BIFF4:

Offset	Size	Contents
0	4	Byte length of the following sheet substream
4	var.	Name of the sheet (byte string, 8-bit string length, →2.1)

5.32 SHEETSOFFSET

BIFF2	BIFF3	BIFF4	BIFF5	BIFF7	BIFF8
---	---	$008E_H$	---	---	---

This record occurs only in BIFF4 workbook files. It is located in the workbook globals section and contains the stream position of the first SHEETHDR record (→5.31). See →4.2 for details about the BIFF4 workbook stream.

Record SHEETSOFFSET, BIFF4:

Offset	Size	Contents
0	4	Stream position of the first SHEETHDR record (→5.31)

5.33 SST – Shared String Table

BIFF2	BIFF3	BIFF4	BIFF5	BIFF7	BIFF8
---	---	---	---	---	$00FC_H$

This record contains a list of all strings used anywhere in the workbook. Each string occurs only one time. The workbook uses indexes into the list to reference to strings. See →4.4 for more information.

Record SST, BIFF8:

Offset	Size	Contents
0	4	Total number of strings in the workbook (see below)
4	4	Number of following strings (nm)
8	var.	List of nm Unicode strings, 16-bit string length (→2.2)

The first field of the SST record counts the total occurrence of strings in the workbook. For instance, the string „AAA“ is used 3 times and the string „BBB“ is used 2 times. The first field contains 5 and the second field contains 2, followed by the two strings.

5.34 STRING

BIFF2	BIFF3	BIFF4	BIFF5	BIFF7	BIFF8
0007_H	0207_H	0207_H	0207_H	0207_H	0207_H

This record stores the result of a string formula. It occurs directly after a string formula (→5.17).

Record STRING, BIFF2:

Offset	Size	Contents
0	var.	Byte string, 8-bit string length (→2.1)

Record STRING, BIFF3-BIFF7:

Offset	Size	Contents
0	var.	Byte string, 16-bit string length (→2.1)

In BIFF8 files the whole record is omitted, if the result is an empty string.

Record STRING, BIFF8:

Offset	Size	Contents
0	var.	Unicode string with at least 1 character, 16-bit string length (→2.2)

5.35 SUPBOOK – External Workbook

BIFF2	BIFF3	BIFF4	BIFF5	BIFF7	BIFF8
---	---	---	---	---	$01AE_H$

This record mainly stores the name of an external document and a list of sheet names inside of this document. Furthermore it is used to store names of documents for DDE and OLE object links or to indicate an internal 3D reference or an AddIn function. See →4.5.3 for details about external references in BIFF8.

5.35.1 External references

A SUPBOOK record for external references stores the name of the document and a list of sheet names.

Record SUPBOOK for external references, BIFF8:

Offset	Size	Contents
0	2	Number of sheet names (nm)
2	var.	Encoded document name without sheet name (→2.6.1). Unicode string, 16-bit string length (→2.2).
var.	var.	List of nm sheet names (Unicode strings with 16-bit string length, →2.2)

5.35.2 Internal references

In each file occurs a SUPBOOK that is used for internal 3D references. It stores the number of sheets of the own document.

Record SUPBOOK for 3D references, BIFF8:

Offset	Size	Contents
0	2	Number of sheets in this document
2	2	0401_H

5.35.3 AddIn functions

AddIn function names are stored in EXTERNNAME records following this SUPBOOK record.

Record SUPBOOK for AddIn functions, BIFF8:

Offset	Size	Contents
0	2	0001_H
2	2	$3A01_H$

5.35.4 DDE links, OLE object links

The SUPBOOK record of a DDE link or an OLE object link contains the name of the server application (DDE) or the class name (OLE) and the name of a source document. These names are encoded in one string.

Record SUPBOOK for DDE links and OLE object links, BIFF8:

Offset	Size	Contents
0	2	0000_H
2	var.	Encoded source document name (→2.6.2). Unicode string, 16-bit string length (→2.2).

5.36 XCT – CRN Count

BIFF2	BIFF3	BIFF4	BIFF5	BIFF7	BIFF8
---	0059_H	0059_H	0059_H	0059_H	0059_H

This record stores the number of immediately following CRN records. These records are used to store the cell contents of external references. See →4.5 for details about of external references.

Record XCT, BIFF3-BIFF7:

Offset	Size	Contents
0	2	Number of following CRN records

Record XCT, BIFF8:

Offset	Size	Contents
0	2	Number of following CRN records
2	2	Index to sheet table of the involved SUPBOOK record (→5.35)

5.37 XF – Extended Format

BIFF2	BIFF3	BIFF4	BIFF5	BIFF7	BIFF8
0043_H	0243_H	0443_H	$00E0_H$	$00E0_H$	$00E0_H$

This record contains formatting information for cells, rows, columns or styles.

From BIFF3 on, some of the elements occur unchanged in every BIFF version. These elements are described in the following using a specific name for each element. In the description of the record structure the names are used to reference to these tables.

5.37.1 XF substructures

• XF_TYPE_PROT – XF type and cell protection (3 bits), BIFF3-BIFF8

These 3 bits are part of a specific data byte.

Bit	Mask	Contents
0	01_H	1 = Cell is locked
1	02_H	1 = Formula is hidden
2	04_H	0 = Cell XF; 1 = Style XF

• XF_USED_ATTRIB – Attributes used from parent style XF (1 byte), BIFF3-BIFF8

In this byte, each bit describes the validity of a specific attribute. In cell XFs an unset bit means the attribute of the parent style XF is used, a set bit means the attribute of this XF is used. In style XFs an unset bit means the attribute setting is valid, a set bit means the attribute should be ignored.

Bit	Mask	Contents
2	04_H	Flag for number format
3	08_H	Flag for font
4	10_H	Flag for alignment, text wrap and rotation
5	20_H	Flag for border lines
6	40_H	Flag for background area style
7	80_H	Flag for cell protection (cell locked and formula hidden)

• XF_HOR_ALIGN – Horizontal alignment (3 bits), BIFF2-BIFF8

The horizontal alignment consists of 3 bits and is part of a specific data byte.

Value	Horizontal alignment
00_H	General
01_H	Left
02_H	Centered
03_H	Right
04_H	Filled
05_H	Justified (BIFF3-BIFF8)
06_H	Centered across selection (BIFF3-BIFF8)

• XF_VERT_ALIGN – Vertical alignment (2 bits), BIFF4-BIFF8

The vertical alignment consists of 2 bits and is part of a specific data byte. Vertical alignment is not available in BIFF2 and BIFF3.

Value	Vertical alignment
00_H	Left
01_H	Centered
02_H	Right
03_H	Justified (BIFF5-BIFF8)

• XF_ORIENTATION – Text orientation (2 bits), BIFF4-BIFF7

In the BIFF versions BIFF4-BIFF7, text can be rotated in steps of 90-degrees or stacked. The orientation mode consists of 2 bits and is part of a specific data byte. In BIFF8 a rotation angle occurs instead of these flags.

Value	Text orientation
00_H	Not rotated
01_H	Letters are stacked top-to-bottom, but not rotated
02_H	Text is rotated 90 degrees counterclockwise
03_H	Text is rotated 90 degrees clockwise

• XF_ROTATION – Text rotation angle (1 byte), BIFF8

Value	Text rotation
0	Not rotated
1-90	1 deg. - 90 deg. counterclockwise
91-180	1 deg. - 90 deg. clockwise
255	Letters are stacked top-to-bottom, but not rotated

• XF_BORDER_34 – Cell border style (4 bytes), BIFF3-BIFF4

Cell borders contain a line style and a line color for each line of the border.

Bit	Mask	Contents
2-0	00000007_H	Top line style (→2.7)
7-3	$000000F8_H$	Index into PALETTE record for top line color (→5.27)
10-8	00000700_H	Left line style
15-11	$0000F800_H$	Index into PALETTE record for left line color
18-16	00070000_H	Bottom line style
23-19	$00F80000_H$	Index into PALETTE record for bottom line color
26-24	07000000_H	Right line style
31-27	$F8000000_H$	Index into PALETTE record for right line color

• XF_AREA_34 – Cell background area style (2 bytes), BIFF3-BIFF4

A cell background area style contains an area pattern and a foreground and background color.

Bit	Mask	Contents
5-0	$003F_H$	Fill pattern (→2.8)
10-6	$07C0_H$	Index into PALETTE record for pattern foreground (→5.27)
15-11	$F800_H$	Index into PALETTE record for pattern background

5.37.2 XF record contents

Record XF, BIFF2:

Offset	Size	Contents		
0	1	Index to FONT record (→5.15)		
1	1	Not used		
2	1	**Bit**	**Mask**	**Contents**
		5-0	$3F_H$	Index to FORMAT record (→5.16)
		6	40_H	1 = Cell is locked
		7	80_H	1 = Formula is hidden
3	1	**Bit**	**Mask**	**Contents**
		2-0	07_H	XF_HOR_ALIGN – Horizontal alignment (see above)
		3	08_H	1 = Cell has left black border
		4	10_H	1 = Cell has right black border
		5	20_H	1 = Cell has top black border
		6	40_H	1 = Cell has bottom black border
		7	80_H	1 = Cell has shaded background

Record XF, BIFF3:

Offset	Size	Contents		
0	1	Index to FONT record (→5.15)		
1	1	Index to FORMAT record (→5.16)		
2	1	XF_TYPE_PROT – XF type and cell protection (see above)		
3	1	XF_USED_ATTRIB – Used attributes (see above)		
4	2	**Bit**	**Mask**	**Contents**
		2-0	0007_H	XF_HOR_ALIGN – Horizontal alignment (see above)
		3	0008_H	1 = Text is wrapped at right border
		15-4	$FFF0_H$	Index to parent style XF (always FFF_H in style XFs)
6	2	XF_AREA_34 – Cell background area (see above)		
8	4	XF_BORDER_34 – Cell border lines (see above)		

Record XF, BIFF4:

Offset	Size	Contents		
0	1	Index to FONT record (→5.15)		
1	1	Index to FORMAT record (→5.16)		
2	2	**Bit**	**Mask**	**Contents**
		2-0	0007_H	XF_TYPE_PROT – XF type, cell protection (see above)
		15-4	$FFF0_H$	Index to parent style XF (always FFF_H in style XFs)
4	1	**Bit**	**Mask**	**Contents**
		2-0	07_H	XF_HOR_ALIGN – Horizontal alignment (see above)
		3	08_H	1 = Text is wrapped at right border
		5-4	30_H	XF_VERT_ALIGN – Vertical alignment (see above)
		7-6	$C0_H$	XF_ORIENTATION – Text orientation (see above)
5	1	XF_USED_ATTRIB – Used attributes (see above)		
6	2	XF_AREA_34 – Cell background area (see above)		
8	4	XF_BORDER_34 – Cell border lines (see above)		

Record XF, BIFF5/BIFF7:

Offset	Size	Contents		
0	2	Index to FONT record (→5.15)		
2	2	Index to FORMAT record (→5.16)		
4	2	**Bit**	**Mask**	**Contents**
		2-0	0007_H	XF_TYPE_PROT – XF type, cell protection (see above)
		15-4	$FFF0_H$	Index to parent style XF (always FFF_H in style XFs)
6	1	**Bit**	**Mask**	**Contents**
		2-0	07_H	XF_HOR_ALIGN – Horizontal alignment (see above)
		3	08_H	1 = Text is wrapped at right border
		5-4	30_H	XF_VERT_ALIGN – Vertical alignment (see above)
7	1	**Bit**	**Mask**	**Contents**
		1-0	03_H	XF_ORIENTATION – Text orientation (see above)
		7-2	FC_H	XF_USED_ATTRIB – Used attributes (see above)
8	4	Cell border lines and background area:		
		Bit	**Mask**	**Contents**
		6-0	$0000007F_H$	Index into PALETTE for pattern foreground
		13-7	$000003F8_H$	Index into PALETTE for pattern background
		21-16	$003F0000_H$	Fill pattern (→2.8)
		24-22	$01C00000_H$	Bottom line style (→2.7)
		31-25	$FE000000_H$	Index into PALETTE for bottom line color
12	4	**Bit**	**Mask**	**Contents**
		2-0	00000007_H	Top line style (→2.7)
		5-3	00000038_H	Left line style
		8-6	$000001C0_H$	Right line style
		15-9	$0000FE00_H$	Index into PALETTE for top line color
		22-16	$007F0000_H$	Index into PALETTE for left line color
		29-23	$3F800000_H$	Index into PALETTE for right line color

Record XF, BIFF8:

Offset	Size	Contents		
0	2	Index to FONT record (→5.15)		
2	2	Index to FORMAT record (→5.16)		
4	2	**Bit**	**Mask**	**Contents**
		2-0	0007_H	XF_TYPE_PROT – XF type, cell protection (see above)
		15-4	$FFF0_H$	Index to parent style XF (always FFF_H in style XFs)
6	1	**Bit**	**Mask**	**Contents**
		2-0	07_H	XF_HOR_ALIGN – Horizontal alignment (see above)
		3	08_H	1 = Text is wrapped at right border
		5-4	30_H	XF_VERT_ALIGN – Vertical alignment (see above)
7	1	XF_ROTATION: Text rotation angle (see above)		
8	1	**Bit**	**Mask**	**Contents**
		3-0	$0F_H$	Indent level
		4	10_H	1 = Shrink content to fit into cell
		5	20_H	1 = Cell is part of a merged range
9	1	**Bit**	**Mask**	**Contents**
		7-2	FC_H	XF_USED_ATTRIB – Used attributes (see above)
10	4	Cell border lines and background area:		
		Bit	**Mask**	**Contents**
		3-0	$0000000F_H$	Left line style (→2.7)
		7-4	$000000F0_H$	Right line style
		11-8	$00000F00_H$	Top line style
		15-12	$0000F000_H$	Bottom line style
		22-16	$007F0000_H$	Index into PALETTE for left line color
		29-23	$3F800000_H$	Index into PALETTE for right line color
		30	40000000_H	1 = Diagonal line from top left to right bottom
		31	80000000_H	1 = Diagonal line from bottom left to right top
14	4	**Bit**	**Mask**	**Contents**
		6-0	$0000007F_H$	Index into PALETTE for top line color
		13-7	$00003F80_H$	Index into PALETTE for bottom line color
		20-14	$001FC000_H$	Index into PALETTE for diagonal line color
		24-21	$01E00000_H$	Diagonal line style (→2.7)
		31-26	$FC000000_H$	Fill pattern (→2.8)
18	2	**Bit**	**Mask**	**Contents**
		6-0	$007F_H$	Index into PALETTE for pattern foreground
		13-7	$3F80_H$	Index into PALETTE for pattern background

6 Drawing Objects, Escher Layer

2do

7 Charts

2do

8 PivotTables

2do

9 Change Tracking

2do