

한글 문서 파일 구조 5.0

Hwp Document File Formats 5.0

revision 1.3:20181108

차 례

저작권	1
본 문서에 대하여... ..	2
I. 한글 5.0 파일 구조	3
1. 개요	5
2. 자료형 설명	6
3. 한글 파일 구조	7
3.1. 한글 파일 구조 요약	7
3.2. 스토리지 별 저장 정보	7
3.2.1. 파일 인식 정보	7
3.2.2. 문서 정보	8
3.2.3. 본문	9
3.2.4. 문서 요약	12
3.2.5. 바이너리 데이터	12
3.2.6. 미리보기 텍스트	12
3.2.7. 미리보기 이미지	13
3.2.8. 문서 옵션	13
3.2.9. 스크립트	13
3.2.10. XML 템플릿	13
3.2.11. 문서 이력 관리	14
3.2.12. 참고문헌	15
4. 데이터 레코드	16
4.1. 데이터 레코드 구조	16
4.2. '문서 정보'의 데이터 레코드	17

4.2.1. 문서 속성	17
4.2.2. 아이디 매핑 헤더	17
4.2.3. 바이너리 데이터	18
4.2.4. 글꼴	20
4.2.5. 테두리/배경	21
4.2.6. 글자 모양	24
4.2.7. 탭 정의	26
4.2.8. 문단 번호	26
4.2.9. 글머리표	27
4.2.10. 문단 모양	28
4.2.11. 스타일(문단 스타일)	29
4.2.12. 문서 임의의 데이터	30
4.2.13. 배포용 문서 데이터	31
4.2.14. 호환 문서	32
4.2.15. 레이아웃 호환성	32
4.3. '본문'의 데이터 레코드	33
4.3.1. 문단 헤더 Tag ID : HWPTAG_PARA_HEADER	33
4.3.2. 문단의 텍스트	34
4.3.3. 문단의 글자 모양	34
4.3.4. 문단의 레이아웃	35
4.3.5. 문단의 영역 태그	35
4.3.6. 컨트롤 헤더	35
4.3.7. 문단 리스트 헤더	36
4.3.8. 컨트롤 임의의 데이터	36
4.3.9. 개체 공통 속성을 포함하는 컨트롤(개체 컨트롤)	36
4.3.9.1. 표 개체	39
4.3.9.2. 그리기 개체(선, 사각형, 타원, 호, 다각형, 곡선)	40
4.3.9.2.1. 개체 요소	40
4.3.9.2.2. 선 개체	42
4.3.9.2.3. 사각형 개체	43
4.3.9.2.4. 타원 개체	43
4.3.9.2.5. 다각형 개체	45
4.3.9.2.6. 호 개체	45

4.3.9.2.7. 곡선 개체	45
4.3.9.3. 한글 수식 개체	46
4.3.9.4. 그림 개체(HWPTAG_SHAPE_COMPONENT_PICTURE)	46
4.3.9.5. OLE 개체(HWPTAG_SHAPE_COMPONENT_OLE)	49
4.3.9.6. 차트 개체	50
4.3.9.7. 묶음 개체(HWPTAG_SHAPE_COMPONENT_CONTAINER)	51
4.3.9.8. 동영상 개체(HWPTAG_VIDEO_TDATA)	51
4.3.10. 개체 이외의 컨트롤	52
4.3.10.1. 구역 정의	53
4.3.10.1.1. 용지 설정	54
4.3.10.1.2. 각주/미주 모양	55
4.3.10.1.3. 쪽 테두리/배경	56
4.3.10.2. 단 정의	56
4.3.10.3. 머리말/꼬리말	57
4.3.10.4. 각주/미주	57
4.3.10.5. 자동 번호	58
4.3.10.6. 새 번호 지정	58
4.3.10.7. 감추기	58
4.3.10.8. 흘/짝수 조정	59
4.3.10.9. 쪽 번호 위치	59
4.3.10.10. 찾아보기 표식	59
4.3.10.11. 책갈피	60
4.3.10.12. 글자 겹침	60
4.3.10.13. 덧말	60
4.3.10.14. 숨은 설명	60
4.3.10.15. 필드 시작	61
4.4. 문서 이력 관리	62
4.4.1. 문서 이력 관리란	62
4.4.2. 문서 이력 관리 레코드 정보	62
4.4.2.1. 히스토리 아이템 정보 시작	62
4.4.2.2. 히스토리 아이템 정보 끝	62
4.4.2.3. 히스토리 아이템 버전	63
4.4.2.4. 히스토리 날짜	63

4.4.2.5. 히스토리 작성자	63
4.4.2.6. 히스토리 설명	63
4.4.2.7. 비교 정보	63
4.4.2.8. 가장 마지막 최근 문서	63
변경 사항 이력	65

저작권

(주)한글과컴퓨터(이하 '한컴')는 문서 형식의 개방성과 표준화에 대하여 적극 찬성합니다. 한컴은 한글 97의 문서 형식을 무상으로 지원한 바 있으며, 한글 2002~2010 문서의 XML 형식은 HWPML에 대해서도 문서 형식을 공개한 바 있습니다. 개방형 문서 표준화 및 코드 관련 위원회에도 적극적으로 참여하여 파일 형식의 표준화와 개방성을 위해 노력해 왔습니다. 이러한 결과로 HWPML 스펙이 OWPML란 이름으로 한국산업표준(KS X 6101:2011)으로 제정되었습니다. 또한, 한컴오피스에서 기록물 장기보존 표준 포맷인 PDF/A-1의 지원과 ISO 국제 문서 형식인 ODF와 OOXML 파일 형식의 불러오기와 저장하기를 적극적으로 지원하였습니다.

본 문서를 열람하고자 하는 자라면 누구에게나 제공되는 것이며, 본 문서를 열람하는 것 외에 복사, 배포, 게재 및 본 문서에 기재되어 있는 내용을 사용하고자 하는 자는 한글과컴퓨터의 본 저작권을 충분히 인식하고 동의하여야 합니다.

본 문서를 누구나 열람, 복사, 배포, 게재 및 사용을 자유롭게 할 수 있습니다. 다만, 배포는 원 내용이 일체 수정되지 않은 원본 또는 복사본으로 제한됩니다. 원본 및 복사본은 한컴에서 제공하는 스펙의 최신 버전을 포함하고 있어야 합니다.

한컴은 한컴오피스 한글 문서 파일(.hwp) 공개 문서에 따라 얻은 결과물을 기초로 또 다른 독점적, 배타적 권리를 취득하고 이를 (주)한글과컴퓨터를 상대로 행사하고자 하는 자를 상대로는 적극적으로 권리행사를 할 수도 있습니다.

그리고, 본 문서 및 본 문서에 기재된 내용을 참고하여 개발한 결과물에 대한 모든 저작권은 결과물을 개발한 개인 또는 단체에 있을 것입니다. 그러나 반드시 개발 결과물에 “본 제품은 한글과컴퓨터의 한글 문서 파일(.hwp) 공개 문서를 참고하여 개발하였습니다.”라고 제품 내 사용자 인터페이스, 매뉴얼, 도움말 및 소스에 모두 기재하여야 하며 제품이 이러한 구성물이 없을 시에는 존재하는 구성물에만 기재합니다. 한컴은 본 문서 및 본 문서에 기재된 내용을 참고하여 개발한 결과물에 대해서 어떠한 정확성, 진실성도 보증하지 아니합니다.

본 문서에 대하여...

본 문서는 한글 워드 프로세서의 파일 저장 형식 중, 한글 2002 이후 제품에서 사용되는 한글 문서 파일 형식 5.0 에 관하여 설명한다.

본 문서는 한글 문서 파일 형식 5.0의 주요한 자료 형식 및 파일 구조, 레코드 구조에 대해서 설명한다.

한글 문서 수식, 차트, 배포용 문서, 한글 문서 파일 형식 3.0, HWPML에 관해서는 별도의 문서에서 설명한다.

I. 한글 5.0 파일 구조

(Hwp Document File Format 5.0)



1. 개요

한글의 문서 파일은 사용자가 따로 지정하지 않는 한 .HWP를 기본 확장자로 가진다. 문서 파일에 저장되는 내용은, 실제 사용자가 입력한 문서의 내용과 문자 장식 정보뿐만 아니라 문서를 편집할 당시의 글꼴에 대한 정보, 조판에 영향을 주는 설정 사항(용지 종류, 여백 정보 등)도 포함된다.

한글 문서 파일 형식 5.0은 2000년 10월 이후에 출시된 한글 제품군(한글 워디안, 한글 2002, 한글 2005, 한글 2007, 한글 2010, 한글 2014, 한글 2018 등)에서 생성되며, 문서 버전에 따라 큰 골격은 유지되나, 추가적인 정보들에 의해 약간의 차이가 있다.

한글 문서 파일 형식 5.0은 파일의 크기를 최소화하기 위하여 압축 기능을 이용한다. 압축된 문서 파일도 기본적인 정보를 저장하는 부분은 압축되지 않으며, 실제 압축되는 부분은 사용자가 입력한 본문과 그림 관련 데이터 부분이다.

한글은 문서 파일의 압축에 zlib.org의 zlib을 사용했다. zlib은 웹상에 소스가 공개되어 있는 공개 소프트웨어이다. zlib은 zlib License를 따르며, 이는 소스의 자유로운 사용이 가능하며 해당 소스를 사용한 2차 산출물에 대한 소스 공개 의무가 없다. 자세한 사항은 zlib에 포함되어 있는 라이선스 문서 파일을 참조하기 바란다.

한글 문서 파일 형식 5.0의 구조는 윈도우즈의 복합 파일(Compound File)에 기초를 두며, 문자 코드는 ISO-10646 표준을 기반으로 한다. 대부분의 문자 정보는 유니코드(UTF-16LE) 형식으로 전달되고, 저장된다.

 Compound File에 대한 접근 방법은 OLE관련 자료 또는 MSDN을 참고
StgOpenStorage(), IStorage::Open(), ...

2. 자료형 설명

앞으로 계속되는 설명에서 한글의 문서 파일에 저장되는 정보는 아래 표에 설명하는 자료형을 이용해 표현한다.

자료형에서 한 바이트는 8 비트로 표현되며, 두 바이트 이상의 길이를 가지는 자료형은 최하위 바이트가 가장 먼저 저장되고, 최상위 바이트가 가장 나중에 저장되는 리틀 엔디언(Little-endian) 형태이다.

파일에 저장되는 자료가 배열(array)일 때는 '자료형 array[개수]'와 같이 표현한다. 예를 들어 10개의 원소를 갖는 word 배열이면 'word array[10]'과 같이 표현한다.

자료형	길이	부호	설명
BYTE	1		부호 없는 한 바이트(0 ~ 255)
WORD	2		16비트 컴파일러에서 'unsigned int'에 해당
DWORD	4		16비트 컴파일러에서 'unsigned long'에 해당
WCHAR	2		한글의 기본 코드로 유니코드 기반 문자
HWPUNIT	4		1/7200인치로 표현된 한글 내부 단위
SHWPUNIT	4	✓	1/7200인치로 표현된 한글 내부 단위
UINT8	1		'unsigned __int8' 에 해당
UINT16	2		'unsigned __int16' 에 해당
UINT32(=UINT)	4		'unsigned __int32' 에 해당
INT8	1	✓	'signed __int8' 에 해당
INT16	2	✓	'signed __int16' 에 해당
INT32	4	✓	'signed __int32' 에 해당
HWPUNIT16	2	✓	INT16 과 같다.
COLORREF	4		RGB값(0x00bbggrr)을 십진수로 표시 (rr : red 1 byte, gg : green 1 byte, bb : blue 1 byte)
BYTE stream			일련의 BYTE로 구성됨. 본문 내에서 다른 구조를 참조할 경우에 사용됨.

표 1 자료형

WCHAR는 한글의 내부 코드로 표현된 문자 한 글자를 표현하는 자료형이다. 한글의 내부 코드는 한글, 영문, 한자를 비롯해 모든 문자가 2 바이트의 일정한 길이를 가진다.

HWPUNIT과 SHWPUNIT는 문자의 크기, 그림의 크기, 용지 여백 등, 문서를 구성하는 요소들의 크기를 표현하기 위한 자료형이다. 문서 출력 장치의 해상도는 가변적이기 때문에, 크기 정보를 점(도트)의 수로 표현할 수는 없다. 따라서 일정한 단위를 기준으로 해야 하는데, 한글에서는 1/7200인치를 기본 단위로 사용한다. 예를 들어 [가로 2 인치 x 세로 1 인치]짜리 그림의 크기를 HWPUNIT 형으로 표현하면 각각 14400 x 7200이 된다.

3. 한글 파일 구조

3.1. 한글 파일 구조 요약

한글의 문서 파일은 개괄적으로 다음 표와 같은 구조를 가진다. 복합 파일(Compound File) 구조를 가지기 때문에, 내부적으로 스토리지(Storage)와 스트림(Stream)을 구별하기 위한 이름을 가진다. 하나의 스트림에는 일반적인 바이너리나 레코드 구조로 데이터가 저장되고, 스트림에 따라서 압축/암호화되기도 한다.

 Storage

 Stream

설명	구별 이름	길이(바이트)	레코드 구조	압축/암호화
파일 인식 정보	FileHeader	고정		
문서 정보	DocInfo	고정	✓	✓
본문	BodyText	가변	✓	✓
	Section0			
	Section1			
	...			
문서 요약	\005HwpSummaryInformation	고정		
바이너리 데이터	BinData	가변		✓
	BinaryData0			
	BinaryData1			
	...			
미리보기 텍스트	PrvText	고정		
미리보기 이미지	PrvImage	가변		
문서 옵션	DocOptions	가변		
	_LinkDoc			
	DrmLicense			
	...			
스크립트	Scripts	가변		
	DefaultJScript			
	JScriptVersion			
	...			
XML 템플릿	XMLTemplate	가변		
	Schema			
	Instance			
	...			
문서 이력 관리	DocHistory	가변	✓	✓
	VersionLog0			
	VersionLog1			
	...			

표 2 전체 구조

압축된 문서 파일의 경우 문서 파일을 읽는 쪽에서는 '파일 인식 정보' 항목의 '압축' 플래그를 살펴보고, 압축된 파일이면 압축을 풀어서 처리해야 한다. 이후의 설명에서는 압축이 풀린 상태의 파일을 기준으로 한다. '문서정보'와 '본문' '문서 이력 관리'에 사용되는 '레코드 구조'는 이후 '데이터 레코드'란에서 구조 설명과 사용되는 레코드들에 대한 상세한 설명을 한다.

3.2. 스토리지 별 저장 정보

3.2.1. 파일 인식 정보

한글의 문서 파일이라는 것을 나타내기 위해 ‘파일 인식 정보’가 저장된다.

FileHeader 스트림에 저장되는 데이터는 다음과 같다.

자료형	길이(바이트)	설명	
BYTE array[32]	32	signature. 문서 파일은 "HWP Document File"	
DWORD	4	파일 버전. 0xMMnnPPrr의 형태(예 5.0.3.0) <ul style="list-style-type: none"> MM: 문서 형식의 구조가 완전히 바뀌는 것을 나타냄. 숫자가 다르면 구 버전과 호환 불가능. nn: 큰 구조는 동일하나, 큰 변화가 있는 것을 나타냄. 숫자가 다르면 구 버전과 호환 불가능. PP: 구조는 동일, Record가 추가되었거나, 하위 버전에서 호환되지 않는 정보가 추가된 것을 나타냄. 숫자가 달라도 구 버전과 호환 가능. rr: Record에 정보들이 추가된 것을 나타냄. 숫자가 달라도 구 버전과 호환 가능. 	
DWORD	4	속성	
		범위	설명
		bit 0	압축 여부
		bit 1	암호 설정 여부
		bit 2	배포용 문서 여부
		bit 3	스크립트 저장 여부
		bit 4	DRM 보안 문서 여부
		bit 5	XMLTemplate 스토리지 존재 여부
		bit 6	문서 이력 관리 존재 여부
		bit 7	전자 서명 정보 존재 여부
		bit 8	공인 인증서 암호화 여부
		bit 9	전자 서명 예비 저장 여부
		bit 10	공인 인증서 DRM 보안 문서 여부
		bit 11	CCL 문서 여부
		bit 12	모바일 최적화 여부
		bit 13	개인 정보 보안 문서 여부
		bit 14	변경 추적 문서 여부
		bit 15	공공누리(KOGL) 저작권 문서
		bit 16	비디오 컨트롤 포함 여부
bit 17	차례 필드 컨트롤 포함 여부		
bit 18 ~ 31	예약		
DWORD	4	속성	
		범위	설명
		bit 0	CCL, 공공누리 라이선스 정보
		bit 1	복제 제한 여부
		bit 2	동일 조건 하에 복제 허가 여부 (복제 제한인 경우 무시)
bit 3 ~ 31	예약		
DWORD	4	EncryptVersion <ul style="list-style-type: none"> 0 : None 1 : (한글 2.5 버전 이하) 2 : (한글 3.0 버전 Enhanced) 3 : (한글 3.0 버전 Old) 4 : (한글 7.0 버전 이후) 	
BYTE	1	공공누리(KOGL) 라이선스 지원 국가 <ul style="list-style-type: none"> 6 : KOR 15 : US 	
BYTE array[207]	207	예약	
전체 길이	256		

표 3 파일 인식 정보

3.2.2. 문서 정보

본문에 사용 중인 글꼴, 글자 속성, 문단 속성, 탭, 스타일 등에 문서 내 공통으로 사용되는 세부 정보를 담고 있다.

DocInfo 스트림에 저장되는 데이터는 다음과 같다.

Tag ID	길이(바이트)	레벨	설명
HWPTAG_DOCUMENT_PROPERTIES	30	0	문서 속성(표 14 참조)
HWPTAG_ID_MAPPINGS	32	0	아이디 매핑 헤더(표 15 참조)
HWPTAG_BIN_DATA	가변	1	바이너리 데이터(표 17 참조)
HWPTAG_FACE_NAME	가변	1	글꼴(표 19 참조)
HWPTAG_BORDER_FILL	가변	1	테두리/배경(표 23 참조)
HWPTAG_CHAR_SHAPE	72	1	글자 모양(표 33 참조)
HWPTAG_TAB_DEF	14	1	탭 정의(표 36 참조)
HWPTAG_NUMBERING	가변	1	문단 번호(표 38 참조)
HWPTAG_BULLET	10	1	글머리표(표 42 참조)
HWPTAG_PARA_SHAPE	54	1	문단 모양(표 43 참조)
HWPTAG_STYLE	가변	1	스타일(표 47 참조)
HWPTAG_MEMO_SHAPE	22	1	메모 모양
HWPTAG_TRACK_CHANGE_AUTHOR	가변	1	변경 추적 작성자
HWPTAG_TRACK_CHANGE	가변	1	변경 추적 내용 및 모양
HWPTAG_DOC_DATA	가변	0	문서 임의의 데이터(표 49 참조)
HWPTAG_FORBIDDEN_CHAR	가변	0	금칙처리 문자
HWPTAG_COMPATIBLE_DOCUMENT	4	0	호환 문서(표 54 참조)
HWPTAG_LAYOUT_COMPATIBILITY	20	1	레이아웃 호환성(표 56 참조)
HWPTAG_DISTRIBUTE_DOC_DATA	256	0	배포용 문서
HWPTAG_TRACKCHANGE	1032	1	변경 추적 정보
전체 길이	가변		

표 4 문서 정보

각각의 세부 정보는 <‘문서 정보’의 데이터 레코드>란에서 추가로 다룬다.

3.2.3. 본문

문서의 본문에 해당되는 문단, 표, 그리기 개체 등의 내용이 저장된다.

BodyText 스토리지는 본문의 구역에 따라 Section%d 스트림(%d는 구역의 번호)으로 구분된다. 구역의 개수는 문서 정보의 문서 속성에 저장된다.

각 구역의 첫 문단에는 구역 정의 레코드가 저장되고, 각 단 설정의 첫 문단에는 단 정의 레코드가 저장된다.

각 구역의 가장 끝 위치에는 확장 바탕쪽(마지막 쪽, 임의 쪽) 관련 정보가 저장되고, 마지막 구역의 가장 끝 위치에는 메모 관련 정보가 저장된다.

Section 스트림에 저장되는 데이터는 문단들(문단 리스트)이며, 다음과 같은 문단 정보들이 반복 된다.

Tag ID	길이(바이트)	레벨	설명
HWPTAG_PARA_HEADER	22	0	문단 헤더(표 58 참조)
HWPTAG_PARA_TEXT	가변	1	문단의 텍스트(표 60 참조)
HWPTAG_PARA_CHAR_SHAPE	가변	1	문단의 글자 모양(표 61 참조)
HWPTAG_PARA_LINE_SEG	가변	1	문단의 레이아웃

HWPTAG_PARA_RANGE_TAG	가변	1	문단의 영역 태그(표 63 참조)
HWPTAG_CTRL_HEADER	4	1	컨트롤 헤더(표 64 참조)
HWPTAG_LIST_HEADER	6	2	문단 리스트 헤더(표 65 참조)
HWPTAG_PAGE_DEF	40	2	용지 설정
HWPTAG_FOOTNOTE_SHAPE	30	2	각주/미주 모양
HWPTAG_PAGE_BORDER_FILL	14	2	쪽 테두리/배경
HWPTAG_SHAPE_COMPONENT	4	2	개체
HWPTAG_TABLE	가변	2	표 개체
HWPTAG_SHAPE_COMPONENT_LINE	20	3	직선 개체
HWPTAG_SHAPE_COMPONENT_RECTANGLE	9	3	사각형 개체
HWPTAG_SHAPE_COMPONENT_ELLIPSE	60	3	타원 개체
HWPTAG_SHAPE_COMPONENT_ARC	25	3	호 개체
HWPTAG_SHAPE_COMPONENT_POLYGON	가변	3	다각형 개체
HWPTAG_SHAPE_COMPONENT_CURVE	가변	3	곡선 개체
HWPTAG_SHAPE_COMPONENT_OLE	26	3	OLE 개체
HWPTAG_SHAPE_COMPONENT_PICTURE	가변	3	그림 개체
HWPTAG_CTRL_DATA	가변	2	컨트롤 임의의 데이터
HWPTAG_EQEDIT	가변	2	수식 개체
HWPTAG_SHAPE_COMPONENT_TEXTART	가변	3	글맵시
HWPTAG_FORM_OBJECT	가변	2	양식 개체
HWPTAG_MEMO_SHAPE	22	1	메모 모양
HWPTAG_MEMO_LIST	4	1	메모 리스트 헤더
HWPTAG_CHART_DATA	2	2	차트 데이터
HWPTAG_VIDEO_DATA	가변	3	비디오 데이터
HWPTAG_SHAPE_COMPONENT_UNKNOWN	36	3	Unknown
전체 길이	가변		

표 5 본문

문단에 컨트롤이 포함되는 경우 컨트롤 헤더 이후로 문단 리스트 헤더와 같은 컨트롤의 레코드 데이터가 저장된다.

□ 제어 문자 (컨트롤)

표, 그림 등 일반 문자로 표현할 수 없는 요소를 표현하기 위해서 문자 코드 중 일부분을 특수 용도로 사용하고 있다.

문단 내용 중에 문자 코드가 0-31인 문자들은 특수 용도로 사용된다. 이미 13번 문자는 문단 내용의 끝 식별 기호로 사용된다는 것은 설명한 바 있다. 이외의 특수 문자들은 표나 그림 등, 일반 문자로 표현할 수 없는 문서 장식 요소를 표현하기 위해서 제어문자(컨트롤)로 사용된다.

제어 문자는 다음 세 가지 형식이 존재한다.

- 문자 컨트롤 [char] = 하나의 문자로 취급되는 문자 컨트롤 / size = 1
- 인라인 컨트롤 [inline] = 별도의 오브젝트 포인터를 가리지 않는 단순한 인라인 컨트롤 / size = 8
- 확장 컨트롤 [extended] = 별도의 오브젝트가 데이터를 표현하는 확장 컨트롤 / size = 8

코드	설명	컨트롤 형식
0	unusable	char
1	예약	extended
2	구역 정의/단 정의	extended
3	필드 시작(누름틀, 하이퍼링크, 블록 책갈피, 표 계산식, 문서 요약, 사용자 정보, 현재 날짜/시간, 문서 날짜/시간, 파일 경로, 상호 참조, 메일 머지, 메모, 교정부호, 개인정보)	extended

4	필드 끝	inline
5-7	예약	inline
8	title mark	inline
9	탭	inline
10	한 줄 끝(line break)	char
11	그리기 개체/표	extended
12	예약	extended
13	문단 끝(para break)	char
14	예약	extended
15	숨은 설명	extended
16	머리말/꼬리말	extended
17	각주/미주	extended
18	자동번호(각주, 표 등)	extended
19-20	예약	inline
21	페이지 컨트롤(감추기, 새 번호로 시작 등)	extended
22	책갈피/찾아보기 표시	extended
23	덧말/글자 겹침	extended
24	하이픈	char
25-29	예약	char
30	묶음 빈칸	char
31	고정폭 빈칸	char

표 6 제어 문자

문서 파일에서 문단 내용을 읽다가 제어 문자를 발견하면, 문서를 읽는 쪽에서는 제어 문자 종류에 따라 읽어 들이거나 건너 뛰어 다음 데이터의 시작 위치까지 파일 포인터를 옮기기 위한 적절한 처리를 수행해야 한다. 제어 문자 가운데는 또 다른 문단 리스트를 포함하는 경우도 있기 때문에, 제어 문자를 일반 문자처럼 처리하면 문서 파일을 정상적으로 읽을 수 없다.

표, 각주 등과 같은 문단 리스트를 포함하는 컨트롤 문자들은 독자적인 문단 리스트를 가진다. 해당 리스트들은 아래와 같은 리스트 헤더 정보를 포함한다. 실제 문단들은 그 다음에 serialize된다.

문단 내에서 컨트롤은 세 가지 형식에 따라 다음과 같은 차이가 있다.

▪ 문자 컨트롤

부가정보 없이 문자 하나로 표현되는 제어 문자이다. (3번째 ch)

0	1	2	3	4	5	6	7	8	9	10	11
'A'	'B'	'C'	ch	'D'	'E'	'F'	'G'	'H'	'I'	'J'	13

▪ 인라인 컨트롤

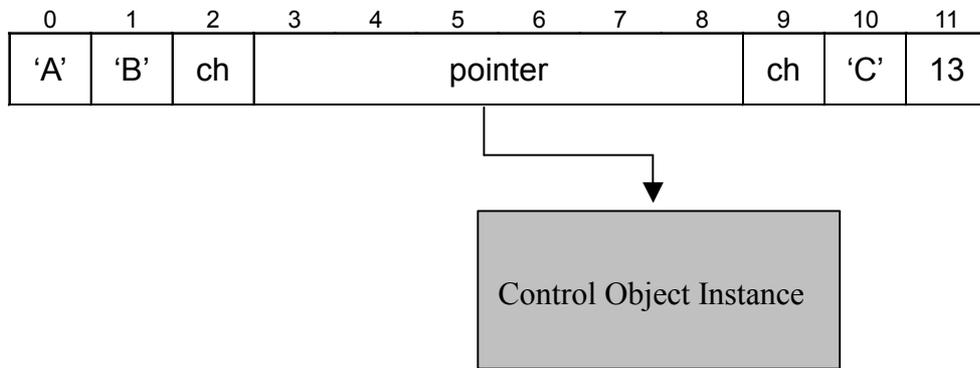
부가정보가 12바이트(6 WCHAR) 이내에서 표현될 수 있는 제어 문자이다. info에 부가정보를 다 넣지 못하는 경우는 확장 컨트롤로 대체된다.(3~9까지 8개의 ch)

0	1	2	3	4	5	6	7	8	9	10	11
'A'	'B'	ch	info						ch	'C'	13

▪ 확장 컨트롤

제어 문자는 포인터를 가지고 있고, 포인터가 가리키는 곳에 실제 오브젝트가 존재하는 제어 문자이

다.(3~9까지 8개의 ch)



본 문서에 부가 설명 없이 '컨트롤' 또는 '제어 문자'이라고 하면 바로 이 확장 컨트롤을 지칭하는 것이다.

3.2.4. 문서 요약

\005HwpSummaryInfomation 스트림에는 한글 메뉴의 “파일-문서 정보-문서 요약”에서 입력한 내용이 저장된다.

- ▶ Summary Information에 대한 자세한 설명은 MSDN을 참고
- The Summary Information Property Set
- The DocumentSummaryInformation and UserDefined Property Set

Name	Property ID string	Property ID	VT type
Title	PIDSI_TITLE	0x00000002	VT_LPSTR
Subject	PIDSI_SUBJECT	0x00000003	VT_LPSTR
Author	PIDSI_AUTHOR	0x00000004	VT_LPSTR
Keywords	PIDSI_KEYWORDS	0x00000005	VT_LPSTR
Comments	PIDSI_COMMENTS	0x00000006	VT_LPSTR
Last Saved By	PIDSI_LASTAUTHOR	0x00000008	VT_LPSTR
Revision Number	PIDSI_REVNUMBER	0x00000009	VT_LPSTR
Last Printed	PIDSI_LASTPRINTED	0x0000000B	VT_FILETIME (UTC)
Create Time/Date(*)	PIDSI_CREATE_DTM	0x0000000C	VT_FILETIME (UTC)
Last saved Time/Date(*)	PIDSI_LASTSAVE_DTM	0x0000000D	VT_FILETIME (UTC)
Number of Pages	PIDSI_PAGECOUNT	0x0000000E	VT_I4
<i>Date String(User define)</i>	<i>HWPPIDSI_DATE_STR</i>	<i>0x00000014</i>	<i>VT_LPSTR</i>
<i>Para Count(User define)</i>	<i>HWPPIDSI_PARACOUNT</i>	<i>0x00000015</i>	<i>VT_I4</i>

표 7 문서 요약

3.2.5. 바이너리 데이터

BinData 스트리지에는 그림이나 OLE 개체와 같이 문서에 첨부된 바이너리 데이터가 각각의 스트림으로 저장된다.

3.2.6. 미리보기 텍스트

PrvText 스트림에는 미리보기 텍스트가 유니코드 문자열로 저장된다.

3.2.7. 미리보기 이미지

PrvImage 스트림에는 미리보기 이미지가 BMP 또는 GIF 형식으로 저장된다.

3.2.8. 문서 옵션

DocOptions 스토리지에는 연결 문서, 배포용 문서, 공인인증서 DRM, 전자 서명 관련 정보가 각각의 스트림으로 저장된다.

_LinkDoc 스트림에는 연결된 문서의 경로가 저장된다.

DrmLicense 스트림에는 DRM Packaging의 Verision 정의가 저장된다.

DrmRootSect 스트림에는 암호화 알고리즘이 저장된다.

CertDrmHeader 스트림에는 DRM Packaging의 Verision 정의가 저장된다.

CertDrmInfo 스트림에는 공인인증서 DRM 정보가 저장된다.

DigitalSignature 스트림에는 전자 서명 정보가 저장된다.

PublicKeyInfo 스트림에는 공개 키 정보가 저장된다.

3.2.9. 스크립트

Scripts 스토리지에는 Script 코드를 저장한다.

JScriptVersion 스트림에는 Script Version이 저장된다.

자료형	길이(바이트)	설명
DWORD	4	스크립트 버전 HIGH
DWORD	4	스크립트 버전 LOW
전체 길이	8	

표 8 스크립트 버전

DefaultJScript 스트림에는 Script 헤더, 소스, Pre 소스, Post 소스가 저장된다.

자료형	길이(바이트)	설명
DWORD	4	스크립트 헤더 길이 (len1)
WCHAR array[len1]	2×len1	스크립트 헤더
DWORD	4	스크립트 소스 길이 (len2)
WCHAR array[len2]	2×len2	스크립트 소스
DWORD	4	스크립트 Pre 소스 길이 (len3)
WCHAR array[len3]	2×len3	스크립트 Pre 소스
DWORD	4	스크립트 Post 소스 길이 (len4)
WCHAR array[len4]	2×len4	스크립트 Post 소스
DWORD	4	스크립트 end flag (-1)
전체 길이	20 + (2×len1) + (2×len2) + (2×len3) + (2×len4)	

표 9 스크립트

3.2.10. XML 템플릿

XMLTemplate 스토리지에는 XML Template 정보를 저장한다.

_SchemaName 스트림에는 Schema 이름 문자열이 저장된다.

자료형	길이(바이트)	설명
DWORD	4	Schema 이름 길이 (len1)
WCHAR array[len1]	2×len1	Schema 이름
전체 길이	4 + (2×len1)	

표 10 Schema 이름 정보

Schema 스트림에는 Schema 문자열이 저장된다.

자료형	길이(바이트)	설명
DWORD	4	Schema 길이 (len1)
WCHAR array[len1]	2×len1	Schema
전체 길이	4 + (2×len1)	

표 11 Schema 길이 정보

Instance 스트림에는 Instance 문자열이 저장된다.

자료형	길이(바이트)	설명
DWORD	4	Instance 길이 (len1)
WCHAR array[len1]	2×len1	Instance
전체 길이	4 + (2×len1)	

표 12 Instance 정보

3.2.11. 문서 이력 관리

한글 메뉴의 “파일-문서 이력 관리”에서 표시 및 생성되는 문서의 이력 정보를 저장하는 장소이다.

문서 이력 정보의 각각의 아이템은 “히스토리” 혹은 “히스토리 아이템”이라 하며 한글 Compound 구조 내에서 각 아이템은 “DocHistory”라는 스토리지 내부에 VersionLog%d(%d는 버전) 이름의 스트림으로 저장된다. 또한, 각각 아이템은 압축, 암호화되어 저장된다.

이력 정보 데이터를 “DocHistory”라는 새로운 스토리지로 저장한다. DocHistory 스토리지에 이력 정보 아이템은 추후 추가 정보가 저장되었을 때 하위 호환성을 위하여 다음의 구조로 저장한다.

DocHistory : 스토리지
VersionLog0 : 이력 정보 스트림
VersionLog1 : 이력 정보 스트림
VersionLog2 : 이력 정보 스트림
⋮
⋮
HistoryLastDoc : 최종문서 스트림

[버전 정보 스토리지 구조]

히스토리 아이템
아이템 시작 레코드
히스토리 정보 레코드 1
히스토리 정보 레코드 2
히스토리 정보 레코드 3
⋮
(사용자, 시간, 날짜등...)
이력 정보 데이터 레코드
아이템 끝 레코드

[히스토리 아이템 스트림 구조]

히스토리 아이템
아이템 시작 레코드
히스토리 정보 레코드 1
히스토리 정보 레코드 2
히스토리 정보 레코드 3
⋮
(사용자, 시간, 날짜등...)
이력 정보 데이터 레코드
아이템 끝 레코드

[히스토리 아이템 스트림 구조]

RECORD HEADER	
자료형	설명
BYTE	레코드 Type
UINT	추가 데이터 블록 (RD) BYTE 크기
RECORD_DATA (RD)	
자료형	설명
DATA	레코드 Type에 따른 다양한 정보 (STRING, Bit flag등)

[레코드 구성]

3.2.12. 참고문헌

Bibliography 스토리지에는 참고문헌 정보가 .XML 파일 형태로 저장한다.

4. 데이터 레코드

4.1. 데이터 레코드 구조

논리적으로 연관된 데이터들을 헤더 정보와 함께 저장하는 방식을 데이터 레코드라고 한다.

레코드 구조를 가지는 스트림은 연속된 여러 개의 레코드로 구성된다. 데이터 레코드는 헤더와 데이터로 구성되며 각 헤더 정보를 활용하여 전체 논리적 구조를 생성하게 된다.

레코드의 헤더에는 데이터 확장에 대비한 정보를 가지고 있다. 따라서 이후에 한글의 기능이 확장되어 레코드에 데이터가 추가되는 경우에도 하위 버전의 한글이 상위 버전의 한글 문서를 읽을 수 있도록 하위 호환성이 보장된다.

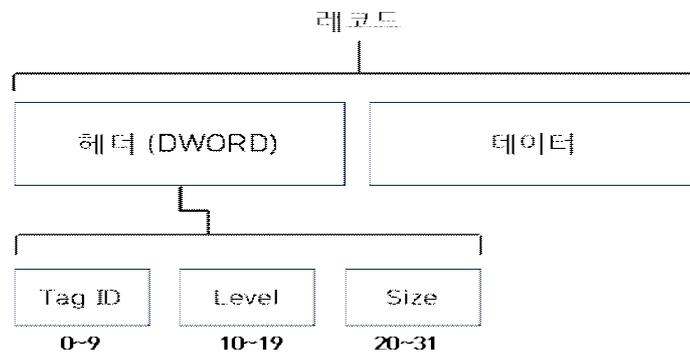


그림 45 레코드 구조

레코드 헤더의 크기는 32bits이고 TagID(10bits), Level(10bits), Size(12bits)로 구성된다.

- **Tag ID** : 레코드가 나타내는 데이터의 종류를 나타내는 태그이다. Tag ID에는 10 비트가 사용되므로 0x000 - 0x3FF까지 가능하다.
 - 0x000 - 0x00F = 일반 레코드 태그가 아닌 특별한 용도로 사용한다.
 - 0x010 - 0x1FF = 한글에 의해 내부용으로 예약된 영역(HWPTAG_BEGIN = 0x010)
 - 0x200 - 0x3FF = 외부 어플리케이션이 사용할 수 있는 영역
- **Level** : 대부분 하나의 오브젝트는 여러 개의 레코드로 구성되는 것이 일반적이기 때문에 하나의 레코드가 아닌 "논리적으로 연관된 연속된 레코드"라는 개념이 필요하다. 레벨은 이와 같이 연관된 레코드의 논리적인 묶음을 표현하기 위한 정보이다. 스트림을 구성하는 모든 레코드는 계층 구조로 표현할 수 있는데, 레벨은 바로 이 계층 구조에서의 depth를 나타낸다.
- **Size** : 데이터 영역의 길이를 바이트 단위로 나타낸다. 12개의 비트가 모두 1일 때는 데이터 영역의 길이가 4095 바이트 이상인 경우로, 이때는 레코드 헤더에 연이어 길이를 나타내는 DWORD가 추가된다. 즉, 4095 바이트 이상의 데이터일 때 레코드는 다음과 같이 표현된다.

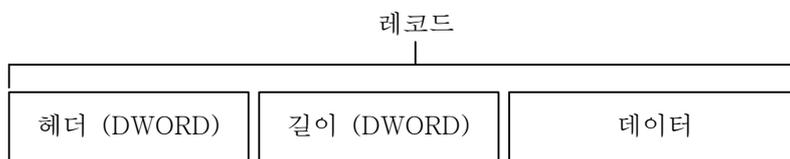


그림 46 확장 데이터 레코드 구조

4.2. '문서 정보'의 데이터 레코드

문서 정보('DocInfo')에서 사용되는 데이터 레코드는 다음과 같다.

Tag ID	Value	의미
HWPTAG_DOCUMENT_PROPERTIES	HWPTAG_BEGIN	문서 속성
HWPTAG_ID_MAPPINGS	HWPTAG_BEGIN+1	아이디 매핑 헤더
HWPTAG_BIN_DATA	HWPTAG_BEGIN+2	BinData
HWPTAG_FACE_NAME	HWPTAG_BEGIN+3	Typeface Name
HWPTAG_BORDER_FILL	HWPTAG_BEGIN+4	테두리/배경
HWPTAG_CHAR_SHAPE	HWPTAG_BEGIN+5	글자 모양
HWPTAG_TAB_DEF	HWPTAG_BEGIN+6	탭 정의
HWPTAG_NUMBERING	HWPTAG_BEGIN+7	번호 정의
HWPTAG_BULLET	HWPTAG_BEGIN+8	불릿 정의
HWPTAG_PARA_SHAPE	HWPTAG_BEGIN+9	문단 모양
HWPTAG_STYLE	HWPTAG_BEGIN+10	스타일
HWPTAG_DOC_DATA	HWPTAG_BEGIN+11	문서의 임의의 데이터
HWPTAG_DISTRIBUTE_DOC_DATA	HWPTAG_BEGIN+12	배포용 문서 데이터
RESERVED	HWPTAG_BEGIN+13	예약
HWPTAG_COMPATIBLE_DOCUMENT	HWPTAG_BEGIN+14	호환 문서
HWPTAG_LAYOUT_COMPATIBILITY	HWPTAG_BEGIN+15	레이아웃 호환성
HWPTAG_TRACKCHANGE	HWPTAG_BEGIN+16	변경 추적 정보
HWPTAG_MEMO_SHAPE	HWPTAG_BEGIN+76	메모 모양
HWPTAG_FORBIDDEN_CHAR	HWPTAG_BEGIN+78	금칙처리 문자
HWPTAG_TRACK_CHANGE	HWPTAG_BEGIN+80	변경 추적 내용 및 모양
HWPTAG_TRACK_CHANGE_AUTHOR	HWPTAG_BEGIN+81	변경 추적 작성자

표 13 문서 정보의 데이터 레코드

4.2.1. 문서 속성

Tag ID : HWPTAG_DOCUMENT_PROPERTIES

자료형	길이(바이트)	설명
UINT16	2	구역 개수
문서 내 각종 시작번호에 대한 정보		
UINT16	2	페이지 시작 번호
UINT16	2	각주 시작 번호
UINT16	2	미주 시작 번호
UINT16	2	그림 시작 번호
UINT16	2	표 시작 번호
UINT16	2	수식 시작 번호
문서 내 캐럿의 위치 정보		
UINT32	4	리스트 아이디
UINT32	4	문단 아이디
UINT32	4	문단 내에서의 글자 단위 위치
전체 길이	26	

표 14 문서 속성

4.2.2. 아이디 매핑 헤더

Tag ID : HWPTAG_ID_MAPPINGS

자료형	길이(바이트)	설명
INT32 array[18]	72	아이디 매핑 개수(표 16 참조)
전체 길이	72	doc version 에 따라 가변적

표 15 아이디 매핑 헤더

값	설명
0	바이너리 데이터
1	한글 글꼴
2	영어 글꼴
3	한자 글꼴
4	일어 글꼴
5	기타 글꼴
6	기호 글꼴
7	사용자 글꼴
8	테두리/배경
9	글자 모양
10	탭 정의
11	문단 번호
12	글머리표
13	문단 모양
14	스타일
15	메모 모양 (5.0.2.1 이상)
16	변경추적 (5.0.3.2 이상)
17	변경추적 사용자 (5.0.3.2 이상)

표 16 아이디 매핑 개수 인덱스

4.2.3. 바이너리 데이터

Tag ID : HWPTAG_BIN_DATA

자료형	길이(바이트)	설명
그림, OLE 등의 바이너리 데이터 아이템에 대한 정보		
UINT16	2	속성(표 18 참조)
WORD	2	Type이 "LINK"일 때, 연결 파일의 절대 경로 길이 (len1)
WCHAR array[len1]	2×len1	Type이 "LINK"일 때, 연결 파일의 절대 경로
WORD	2	Type이 "LINK"일 때, 연결 파일의 상대 경로 길이 (len2)
WCHAR array[len2]	2×len2	Type이 "LINK"일 때, 연결 파일의 상대 경로
UINT16	2	Type이 "EMBEDDING"이거나 "STORAGE"일 때, BINDATASTORAGE에 저장된 바이너리 데이터의 아이디
WORD	2	Type이 "EMBEDDING"일 때, 바이너리 데이터의 형식 이름의 길이 (len3)
WCHAR array[len3]	2×len3	Type이 "EMBEDDING"일 때 extension(".") 제외
		그림의 경우
		ole
		OLE의 경우
전체 길이	가변	10 + (2×len1) + (2×len2) + (2×len3) 바이트

표 17 바이너리 데이터

범위	구분	값	설명
bit 0~3	Type	0x0000	LINK. 그림 외부 파일 참조
		0x0001	EMBEDDING. 그림 파일 포함
		0x0002	STORAGE. OLE 포함
bit 4~5	압축	0x0000	스토리지의 디폴트 모드 따라감
		0x0010	무조건 압축
		0x0020	무조건 압축하지 않음
bit 8~9	상태	0x0000	아직 access 된 적이 없는 상태
		0x0100	access에 성공하여 파일을 찾은 상태
		0x0200	access가 실패한 에러 상태
		0x0300	링크 access가 실패했으나 무시된 상태

표 18 바이너리 데이터 속성

4.2.4. 글꼴

Tag ID : HWPTAG_FACE_NAME

자료형	길이(바이트)	설명
BYTE	1	속성(표 15 참조)
WORD	2	글꼴 이름 길이(len1)
WCHAR array[len1]	2×len1	글꼴 이름
BYTE	1	대체 글꼴 유형(표 16 참조)
WORD	2	대체 글꼴 이름 길이(len2)
WCHAR array[len2]	2×len2	대체 글꼴 이름
BYTE array[10]	10	글꼴 유형 정보(표 17 참조)
WORD	2	기본 글꼴 이름 길이(len3)
WCHAR array[len3]	2×len3	기본 글꼴 이름
전체 길이	가변	18 + (2×len1) + (2×len2) + (2×len3) 바이트

표 19 글꼴

값	설명
0x80	대체 글꼴 존재 여부
0x40	글꼴 유형 정보 존재 여부
0x20	기본 글꼴 존재 여부

표 20 글꼴 속성

값	설명
0	원래 종류를 알 수 없을 때
1	트루타입 글꼴(TTF)
2	한글 전용 글꼴(HFT)

표 21 대체 글꼴 유형

자료형	길이(바이트)	설명
BYTE	1	글꼴 계열
BYTE	1	세리프 유형
BYTE	1	굵기
BYTE	1	비례
BYTE	1	대조
BYTE	1	스트로크 편차
BYTE	1	자획 유형
BYTE	1	글자형
BYTE	1	중간선
BYTE	1	X-높이

표 22 글꼴 유형 정보

4.2.5. 테두리/배경

Tag ID : HWPTAG_BORDER_FILL

자료형	길이(바이트)	설명
UINT16	2	속성(표 24 참조)
UINT8 array[4]	4	4방향 테두리선 종류(표 25 참조)
UINT8 array[4]	4	4방향 테두리선 굵기(표 26 참조)
COLORREF array[4]	16	4방향 테두리선 색상.
UINT8	1	대각선 종류(표 27 참조)
UINT8	1	대각선 굵기
COLORREF	4	대각선 색깔
BYTE stream	n	채우기 정보(표 28 참조)
전체 길이	가변	32+n 바이트

표 23 테두리/배경 속성

범위	설명
bit 0	3D 효과의 유무
bit 1	그림자 효과의 유무
bit 2~4	Slash 대각선 모양(시계 방향으로 각각의 대각선 유무를 나타냄) 000 : none 010 : slash 011 : LeftTop --> Bottom Edge 110 : LeftTop --> Right Edge 111 : LeftTop --> Bottom & Right Edge
bit 5~7	BackSlash 대각선 모양(반시계 방향으로 각각의 대각선 유무를 나타냄) 000 : none 010 : / back slash 011 : RightTop --> Bottom Edge 110 : RightTop --> Left Edge 111 : RightTop --> Bottom & Left Edge
bit 8~9	Slash 대각선 꺾은선
bit 10	BackSlash 대각선 꺾은선
bit 11	Slash 대각선 모양 180도 회전 여부
bit 12	BackSlash 대각선 모양 180도 회전 여부
bit 13	중심선 유무.

표 24 테두리/배경 속성

값	설명
0	실선
1	긴 점선
2	점선
3	-.-.-. .
4	-.-.-. .
5	Dash보다 긴 선분의 반복
6	Dot보다 큰 동그라미의 반복
7	2중선
8	가는선 + 굵은선 2중선
9	굵은선 + 가는선 2중선
10	가는선 + 굵은선 + 가는선 3중선
11	물결
12	물결 2중선
13	두꺼운 3D

14	두꺼운 3D(광원 반대)
15	3D 단선
16	3D 단선(광원 반대)

표 25 테두리선 종류

값	설명	값	설명
0	0.1 mm	8	0.6 mm
1	0.12 mm	9	0.7 mm
2	0.15 mm	10	1.0 mm
3	0.2 mm	11	1.5 mm
4	0.25 mm	12	2.0 mm
5	0.3 mm	13	3.0 mm
6	0.4 mm	14	4.0 mm
7	0.5 mm	15	5.0 mm

표 26 테두리선 굵기

값	설명
0	Slash
1	BackSlash
2	CrookedSlash

표 27 대각선 종류

자료형	길이(바이트)	설명
UINT	4	채우기 종류(type) 0x00000000 : 채우기 없음 0x00000001 : 단색 채우기 0x00000002 : 이미지 채우기 0x00000004 : 그레데이션 채우기
단색 채우기 (type & 0x00000001 != 0)		
COLORREF	4	배경색
COLORREF	4	무늬색
INT32	4	무늬 종류(표 29 참조)
그레데이션 채우기 (type & 0x00000004 != 0)		
INT16	2	그레데이션 유형(표 30 참조)
INT16	2	그레데이션의 기울임(시작 각)
INT16	2	그레데이션의 가로 중심(중심 X 좌표)
INT16	2	그레데이션의 세로 중심(중심 Y 좌표)
INT16	2	그레데이션 번짐 정도(0 -100)
INT16	2	그레데이션의 색 수(num). (워드안/한글2002/SE에서는 항상 2이다.)
INT32	4×num	색상이 바뀌는 곳의 위치. (num > 2 일 경우에만)
COLORREF array[num]	4×num	색상
이미지 채우기 (type & 0x00000002 != 0)		
BYTE	1	이미지 채우기 유형(표 31 참조)
BYTE stream	5	그림 정보(표 32 참조)
DWORD	4	type의 0x0004비트가 켜져 있으면 size(=추가적인 그레데이션 바이트) = 1, 아니면 size = 0
BYTE	1	그레데이션 번짐정도의 중심 (0..100)
DWORD	4	추가 채우기 속성 길이(size)
BYTE	size	추가 채우기 속성 - 그레데이션일 경우(type & 0x00000001 != 0) 번짐 정도의 중심(0~100) : 1byte
전체 길이	가변	8 + n 바이트

표 28 채우기 정보

값	설명
1	-----
2	
3	\\ \\
4	////
5	+++++
6	xxxxx

표 29 채우기 무늬 종류

값	설명
1	출무늬형
2	원형
3	원뿔형
4	사각형

표 30 그레데이션 유형

값	설명
0	바둑판식으로-모두
1	바둑판식으로-가로/위
2	바둑판식으로-가로/아래
3	바둑판식으로-세로/왼쪽
4	바둑판식으로-세로/오른쪽
5	크기에 맞추어
6	가운데로
7	가운데 위로
8	가운데 아래로
9	왼쪽 가운데로
10	왼쪽 위로
11	왼쪽 아래로
12	오른쪽 가운데로
13	오른쪽 위로
14	오른쪽 아래로
15	NONE

표 31 이미지 채우기 유형

자료형	길이(바이트)	설명	
INT8	1	밝기	
INT8	1	명암	
BYTE	1	그림 효과	
		값	설명
		0	REAL_PIC
		1	GRAY_SCALE
		2	BLACK_WHITE
4	PATTERN8x8		
UINT16	2	BinItem의 아이디 참조값	
전체 길이	5		

표 32 그림 정보

4.2.6. 글자 모양

Tag ID : HWPTAG_CHAR_SHAPE

자료형	길이(바이트)	설명
WORD array[7]	14	언어별 글꼴 ID(FaceID) 참조 값(표 34 참조)
UINT8 array[7]	7	언어별 장평, 50% ~ 200%(표 34 참조)
INT8 array[7]	7	언어별 자간, -50% ~ 50%(표 34 참조)
UINT8 array[7]	7	언어별 상대 크기, 10% ~ 250%(표 34 참조)
INT8 array[7]	7	언어별 글자 위치, -100% ~ 100%(표 34 참조)
INT32	4	기준 크기, Opt ~ 4096pt
UINT32	4	속성(표 30 참조)
INT8	1	그림자 간격, -100% ~ 100%
INT8	1	그림자 간격, -100% ~ 100%
COLORREF	4	글자 색
COLORREF	4	밑줄 색
COLORREF	4	음영 색
COLORREF	4	그림자 색
UINT16	2	글자 테두리/배경 ID(CharShapeBorderFill ID) 참조 값 (5.0.2.1 이상)
COLORREF	4	취소선 색 (5.0.3.0 이상)
전체 길이	72	

표 33 글자 모양

값	설명
0	한글
1	영어
2	한자
3	일어
4	기타
5	기호
6	사용자

표 34 글꼴에 대한 언어

범위	구분	값	설명
bit 0	기울임 여부		
bit 1	진하게 여부		
bit 2~3	밑줄 종류	0	없음
		1	글자 아래
		3	글자 위
bit 4~7	밑줄 모양		(표 25 참조)
bit 8~10	외곽선 종류	0	없음
		1	실선
		2	점선
		3	굵은 실선(두꺼운 선)
		4	파선(긴 점선)
		5	일점쇄선 (-.-.-.)
		6	이점쇄선 (-.-.-.)
bit 11~12	그림자 종류	0	없음
		1	비연속
		2	연속
bit 13	양각 여부		
bit 14	음각 여부		
bit 15	위 첨자 여부		
bit 16	아래 첨자 여부		
bit 17	Reserved.		
bit 18~20	취소선 여부		
bit 21~24	강조점 종류	0	없음
		1	검정 동그라미 강조점
		2	속 빈 동그라미 강조점
		3	~
		4	~
		5	.
		6	:
bit 25	글꼴에 어울리는 빈칸 사용 여부		
bit 26~29	취소선 모양		(표 25 참조)
bit 30	Kerning 여부		

표 35 글자 모양 속성

4.2.7. 탭 정의

Tag ID : HWPTAG_TAB_DEF

자료형	길이(바이트)	설명	
UINT32	4	속성	
INT16	4	count	
탭 정보 (count 개수)			
HPWUNIT	4	탭의 위치	
UINT8	1	탭의 종류	
		값	설명
		0	왼쪽
		1	오른쪽
		2	가운데
3	소수점		
UINT8	1	채움 종류(표 25 참조)	
UINT16	2	8 바이트를 맞추기 위한 예약	
전체 길이	8+(8×count)		

표 36 탭 정의

범위	설명
bit 0	문단 왼쪽 끝 자동 탭(내어 쓰기용 자동 탭) 유무
bit 1	문단 오른쪽 끝 자동 탭 유무

표 37 탭 정의 속성

4.2.8. 문단 번호

Tag ID : HWPTAG_NUMBERING

자료형	길이(바이트)	설명	
BYTE stream	8	문단 머리 정보(표 39 참조)	
WORD	2	번호 형식 길이 (len)	
WCHAR array[len]	2×len	번호 형식. 볼릿 문단의 경우에는 사용되지 않는다. 문자열 내 특정 문자에 제어코드(^)를 붙임으로써 한글에서 표시되는 번호 문단 머리의 형식을 제어한다. ^n : 레벨 경로를 표시한다.(예: 1.1.1.1.1.1.1) ^N : 레벨 경로를 표시하며 마지막에 마침표를 하나 더 찍는다.(예: 1.1.1.1.1.1.1.) 표 36 참조)	7회 반복 수준(1~7) 각 레벨에 해당하는 숫자 또는 문자 또는 기호를 표시
UINT16	2	시작 번호	
UINT	4	수준별 시작번호 (5.0.2.5 이상)	
WORD	2	확장 번호 형식 길이 (len)	
WCHAR array[len]	2×len	확장 번호 형식. 볼릿 문단의 경우에는 사용되지 않는다. 문자열 내 특정 문자에 제어코드(^)를 붙임으로써 한글에서 표시되는 번호 문단 머리의 형식을 제어한다. ^n : 레벨 경로를 표시한다.(예:	3회 반복 수준(8~10) 각 레벨에 해당하는 숫자 또는 문자 또는 기호를 표시

		1.1.1.1.1.1.1) ^N : 레벨 경로를 표시하며 마지막에 마침표를 하나 더 찍는다.(예: 1.1.1.1.1.1.1.) 표 36 참조)	
UINT	4	확장 수준별 시작번호 (5.1.0.0 이상)	수준(8~10)
전체 길이	가변	10 + (2×len)×7 + 2 + (4×7) (2×len)×3 + (4×3) 바이트	

표 38 문단 번호

자료형	길이(바이트)	설명
UINT	4	속성(표 40 참조)
HWPUNIT16	2	너비 보정값
HWPUNIT16	2	본문과의 거리
UINT	4	글자 모양 아이디 참조
전체 길이	8	

표 39 문단 머리 정보

범위	구분	값	설명
bit 0~1	문단의 정렬 종류	0	왼쪽
		1	가운데
		2	오른쪽
bit 2	번호 너비를 실제 인스턴스 문자열의 너비에 따를지 여부	0	false
		1	true
bit 3	자동 내어 쓰기 여부	0	false
		1	true
bit 4	수준별 본문과의 거리 종류	0	글자 크기에 대한 상대 비율
		1	값

표 40 문단 머리 정보 속성

값	설명
0	1, 2, 3
1	동그라미 쳐진 1, 2, 3
2	I, II, III
3	i, ii, iii
4	A, B, C
5	a, b, c
6	동그라미 쳐진 A, B, C
7	동그라미 쳐진 a, b, c
8	가, 나, 다
9	동그라미 쳐진 가, 나, 다
10	ㄱ, ㄴ, ㄷ
11	동그라미 쳐진 ㄱ, ㄴ, ㄷ
12	일, 이, 삼
13	一, 二, 三
14	동그라미 쳐진 一, 二, 三

표 41 문단 번호 형식

4.2.9. 글머리표

Tag ID : HWPTAG_BULLET

자료형	길이(바이트)	설명
BYTE stream	8	문단 머리의 정보
WCHAR	2	글머리표 문자
INT32	4	이미지 글머리표 여부 (글머리표 :0, 이미지글머리표 : ID)
BYTE stream	4	이미지 글머리 (대비, 밝기 ,효과, ID)
WCHAR	2	체크 글머리표 문자
전체 길이	20	

표 42 글머리표

4.2.10. 문단 모양

Tag ID : HWPTAG_PARA_SHAPE

자료형	길이(바이트)	설명
UINT32	4	속성 1(표 44 참조)
INT32	4	왼쪽 여백
INT32	4	오른쪽 여백
INT32	4	들어 쓰기/내어 쓰기
INT32	4	문단 간격 위
INT32	4	문단 간격 아래
INT32	4	줄 간격. 한글 2007 이하 버전(5.0.2.5 버전 미만)에서 사용.
UINT16	2	탭 정의 아이디(TabDef ID) 참조 값
UINT16	2	번호 문단 ID(Numbering ID) 또는 글머리표 문단 모양 ID(Bullet ID) 참조 값
UINT16	2	테두리/배경 모양 ID(BorderFill ID) 참조 값
INT16	2	문단 테두리 왼쪽 간격
INT16	2	문단 테두리 오른쪽 간격
INT16	2	문단 테두리 위쪽 간격
INT16	2	문단 테두리 아래쪽 간격
UINT32	4	속성 2(표 40 참조) (5.0.1.7 버전 이상)
UINT32	4	속성 3(표 41 참조) (5.0.2.5 버전 이상)
UINT32	4	줄 간격(5.0.2.5 버전 이상)
전체 길이	54	

표 43 문단 모양

범위	구분	값	설명
bit 0~1	줄 간격 종류. 한글 2007 이하 버전에서 사용.	0	글자에 따라(%)
		1	고정값
		2	여백만 지정
bit 2~4	정렬 방식	0	양쪽 정렬
		1	왼쪽 정렬
		2	오른쪽 정렬
		3	가운데 정렬
		4	배분 정렬
		5	나눔 정렬
bit 5~6	줄 나눔 기준 영어 단위	0	단어
		1	하이픈
		2	글자
bit 7	줄 나눔 기준 한글 단위	0	어절
		1	글자
bit 8	편집 용지의 줄 격자 사용 여부		

bit 9~15	공백 최소값	0%~75%	
bit 16	외톨이줄 보호 여부		
bit 17	다음 문단과 함께 여부		
bit 18	문단 보호 여부		
bit 19	문단 앞에서 항상 쪽 나눔 여부		
bit 20~21	세로 정렬	0	글꼴기준
		1	위쪽
		2	가운데
		3	아래
bit 22	글꼴에 어울리는 줄 높이 여부		
bit 23~24	문단 머리 모양 종류	0	없음
		1	개요
		2	번호
		3	글머리표(bullet)
bit 25~27	문단 수준	1수준~7수준	
bit 28	문단 테두리 연결 여부		
bit 29	문단 여백 무시 여부		
bit 30	문단 꼬리 모양		

표 44 문단 모양 속성1

범위	구분	값	설명
bit 0~1	한 줄로 입력 여부		
bit 2~3	Reserved		
bit 4	한글과 영어 간격을 자동 조절 여부		
bit 5	한글과 숫자 간격을 자동 조절 여부		

표 45 문단 모양 속성2

범위	구분	값	설명
bit 0~4	줄 간격 종류	0	글자에 따라
		1	고정 값
		2	여백만 지정
		3	최소

표 46 줄 간격 종류

4.2.11. 스타일(문단 스타일)

Tag ID : HWPTAG_STYLE

자료형	길이(바이트)	설명
WORD	2	길이 (len1)
WCHAR array[len1]	2×len1	로컬 스타일 이름. 한글 윈도우에서는 한글 스타일 이름
WORD	2	길이 (len2)
WCHAR array[len2]	2×len2	영문 스타일 이름
BYTE	1	속성(표 43 참조)
BYTE	1	다음 스타일 아이디 참조값
INT16	2	언어 아이디(표 48 참조)
UINT16	2	문단 모양 아이디 참조값(문단 모양의 아이디 속성) 스타일의 종류가 문단인 경우 반드시 지정해야 한다.
UINT16	2	글자 모양 아이디(글자 모양의 아이디 속성) 스타일의 종류가 글자인 경우 반드시 지정해야 한다.

전체 길이	가변	12 + (2×len1) + (2×len2) 바이트
-------	----	------------------------------

표 47 스타일

범위	구분	값	설명
bit 0~2	스타일 종류	0	문단 스타일
		1	글자 스타일

표 48 스타일 종류

4.2.12. 문서 임의의 데이터

라벨 문서인지 여부나 인쇄 대화상자의 정보를 저장한다.

Tag ID : HWPTAG_DOC_DATA

자료형	길이(바이트)	설명
Parameter Set	가변	파라미터 셋(표 50 참조)
전체 길이	가변	

표 49 문서 임의의 데이터

파라미터 아이템의 개수만큼 아이템 데이터를 얻는다.

자료형	길이(바이트)	설명
WORD	2	파라미터 셋 ID
INT16	2	파라미터 셋에 존재하는 아이템 개수(n)
Parameter Item	가변×n	파라미터 아이템(표 51 참조)
전체 길이	가변	4 + (가변×n) 바이트

표 50 파라미터 셋

자료형	길이(바이트)	설명
WORD	2	파라미터 아이템 ID
WORD	2	파라미터 아이템 종류(표 52 참조)
Parameter Item Type	가변	파라미터 아이템 데이터
전체 길이	가변	4 + 가변 바이트

표 51 파라미터 아이템

값	구분	자료형	설명
0	PIT_NULL	UINT	NULL
1	PIT_BSTR	WORD	문자열 길이(slen)
		WCHAR array[len]	문자열
2	PIT_I1	UINT	INT8
3	PIT_I2	UINT	INT16
4	PIT_I4	UINT	INT32
5	PIT_I	UINT	INT
6	PIT_UI1	UINT	UINT8
7	PIT_UI2	UINT	UINT16
8	PIT_UI4	UINT	UINT32
9	PIT_UI	UINT	UINT
0x8000	PIT_SET	Parameter Set	파라미터 셋
0x8001	PIT_ARRAY	INT16	파라미터 셋 개수
		ParameterArray	파라미터 셋 배열
0x8002	PIT_BINDATA	UINT16	바이너리 데이터 ID

표 52 파라미터 아이템 종류

4.2.13. 배포용 문서 데이터

배포용 문서에서는 모든 스트림에 배포용 문서 데이터가 들어간다.

Tag ID : HWPTAG_DISTRIBUTE_DOC_DATA

자료형	길이(바이트)	설명
BYTE array[256]	256	배포용 문서 데이터
전체 길이		

표 53 배포용 문서 데이터

4.2.14. 호환 문서

Tag ID : HWPTAG_COMPATIBLE_DOCUMENT

자료형	길이(바이트)	설명
UINT32	4	대상 프로그램(표 55 참조)
전체 길이	4	

표 54 호환 문서

값	설명
0	한글 문서(현재 버전)
1	한글 2007 호환 문서
2	MS 워드 호환 문서

표 55 대상 프로그램

4.2.15. 레이아웃 호환성

Tag ID : HWPTAG_LAYOUT_COMPATIBILITY

자료형	길이(바이트)	설명
UINT32	4	글자 단위 서식
UINT32	4	문단 단위 서식
UINT32	4	구역 단위 서식
UINT32	4	개체 단위 서식
UINT32	4	필드 단위 서식
전체 길이	20	

표 56 레이아웃 호환성

4.3. '본문'의 데이터 레코드

본문에서 사용되는 데이터 레코드는 다음과 같다.

Tag ID	Value	설명
HWPTAG_PARA_HEADER	HWPTAG_BEGIN+50	문단 헤더
HWPTAG_PARA_TEXT	HWPTAG_BEGIN+51	문단의 텍스트
HWPTAG_PARA_CHAR_SHAPE	HWPTAG_BEGIN+52	문단의 글자 모양
HWPTAG_PARA_LINE_SEG	HWPTAG_BEGIN+53	문단의 레이아웃
HWPTAG_PARA_RANGE_TAG	HWPTAG_BEGIN+54	문단의 영역 태그
HWPTAG_CTRL_HEADER	HWPTAG_BEGIN+55	컨트롤 헤더
HWPTAG_LIST_HEADER	HWPTAG_BEGIN+56	문단 리스트 헤더
HWPTAG_PAGE_DEF	HWPTAG_BEGIN+57	용지 설정
HWPTAG_FOOTNOTE_SHAPE	HWPTAG_BEGIN+58	각주/미주 모양
HWPTAG_PAGE_BORDER_FILL	HWPTAG_BEGIN+59	쪽 테두리/배경
HWPTAG_SHAPE_COMPONENT	HWPTAG_BEGIN+60	개체
HWPTAG_TABLE	HWPTAG_BEGIN+61	표 개체
HWPTAG_SHAPE_COMPONENT_LINE	HWPTAG_BEGIN+62	직선 개체
HWPTAG_SHAPE_COMPONENT_RECTANGLE	HWPTAG_BEGIN+63	사각형 개체
HWPTAG_SHAPE_COMPONENT_ELLIPSE	HWPTAG_BEGIN+64	타원 개체
HWPTAG_SHAPE_COMPONENT_ARC	HWPTAG_BEGIN+65	호 개체
HWPTAG_SHAPE_COMPONENT_POLYGON	HWPTAG_BEGIN+66	다각형 개체
HWPTAG_SHAPE_COMPONENT_CURVE	HWPTAG_BEGIN+67	곡선 개체
HWPTAG_SHAPE_COMPONENT_OLE	HWPTAG_BEGIN+68	OLE 개체
HWPTAG_SHAPE_COMPONENT_PICTURE	HWPTAG_BEGIN+69	그림 개체
HWPTAG_SHAPE_COMPONENT_CONTAINER	HWPTAG_BEGIN+70	컨테이너 개체
HWPTAG_CTRL_DATA	HWPTAG_BEGIN+71	컨트롤 임의의 데이터
HWPTAG_EQEDIT	HWPTAG_BEGIN+72	수식 개체
RESERVED	HWPTAG_BEGIN+73	예약
HWPTAG_SHAPE_COMPONENT_TEXTART	HWPTAG_BEGIN+74	글맵시
HWPTAG_FORM_OBJECT	HWPTAG_BEGIN+75	양식 개체
HWPTAG_MEMO_SHAPE	HWPTAG_BEGIN+76	메모 모양
HWPTAG_MEMO_LIST	HWPTAG_BEGIN+77	메모 리스트 헤더
HWPTAG_CHART_DATA	HWPTAG_BEGIN+79	차트 데이터
HWPTAG_VIDEO_DATA	HWPTAG_BEGIN+82	비디오 데이터
HWPTAG_SHAPE_COMPONENT_UNKNOWN	HWPTAG_BEGIN+99	Unknown

표 57 본문의 데이터 레코드

4.3.1. 문단 헤더 Tag ID : HWPTAG_PARA_HEADER

자료형	길이(바이트)	설명
UINT32	4	text(=chars) if (nchars & 0x80000000) { nchars &= 0x7fffffff; }
UINT32	4	control mask (UINT32)(1<<ctrlch) 조합 ctrlch는 HwpCtrlAPI.Hwp 2.1 CtrlCh 참고
UINT16	2	문단 모양 아이디 참조값
UINT8	1	문단 스타일 아이디 참조값
UINT8	1	단 나누기 종류(표 59 참조)
UINT16	2	글자 모양 정보 수

UINT16	2	range tag 정보 수
UINT16	2	각 줄에 대한 align에 대한 정보 수
UINT32	4	문단 Instance ID (unique ID)
UINT16	2	변경추적 병합 문단여부. (5.0.3.2 버전 이상)
전체 길이	24	

표 58 문단 헤더

값	설명
0x01	구역 나누기
0x02	다단 나누기
0x04	쪽 나누기
0x08	단 나누기

표 59 단 나누기 종류

텍스트의 수가 1 이상이면 문자 수만큼 텍스트를 로드하고 그렇지 않을 경우 PARA_BREAK로 문단을 생성한다.

4.3.2. 문단의 텍스트

Tag ID : HWPTAG_PARA_TEXT

자료형	길이(바이트)	설명
WCHAR array[sizeof(nchars)]	2×nchars	문자수만큼의 텍스트
전체 길이	가변	(2×nchars) 바이트

표 60 문단 텍스트

문단은 최소 하나의 문자 Shape buffer가 존재하며, 첫 번째 pos가 반드시 0이어야 한다.

예를 들어 문자길이 40자 짜리 문단이 10자씩 4가지 다른 글자 모양으로 구성되어 있다면 buffer는 다음과 같다.

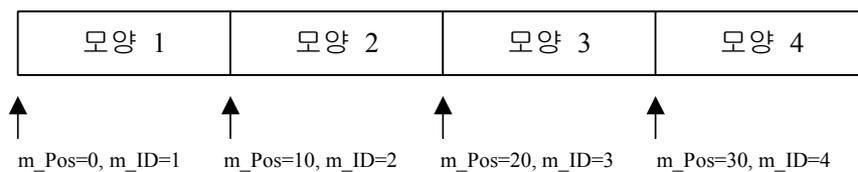


그림 47 문단 버퍼 구조

텍스트 문자 Shape 레코드를 글자 모양 정보 수(Character Shapes)만큼 읽는다.

4.3.3. 문단의 글자 모양

Tag ID : HWPTAG_PARA_CHAR_SHAPE

자료형	길이(바이트)	설명
UINT32	4	글자 모양이 바뀌는 시작 위치
UINT32	4	글자 모양 ID
전체 길이	가변	8×n

표 61 문단의 글자 모양

4.3.4. 문단의 레이아웃

문단의 각 줄을 출력할 때 사용한 Cache 정보이며, 문단 정보의 '각 줄에 대한 align에 대한 정보 수'만큼 반복한다.

Tag ID : HWPTAG_PARA_LINE_SEG

자료형	길이(바이트)	설명
UINT32	4	텍스트 시작 위치
INT32	4	줄의 세로 위치
INT32	4	줄의 높이
INT32	4	텍스트 부분의 높이
INT32	4	줄의 세로 위치에서 베이스라인까지 거리
INT32	4	줄간격
INT32	4	컬럼에서의 시작 위치
INT32	4	세그먼트의 폭
UINT32	4	태그 -bit 0 : 페이지의 첫 줄인지 여부 -bit 1 : 컬럼의 첫 줄인지 여부 -bit 16 : 텍스트가 배열되지 않은 빈 세그먼트인지 여부 -bit 17 : 줄의 첫 세그먼트인지 여부 -bit 18 : 줄의 마지막 세그먼트인지 여부 -bit 19 : 줄의 마지막에 auto-hyphenation이 수행되었는지 여부. -bit 20 : indentation 적용 -bit 21 : 문단 머리 모양 적용 -bit 31 : 구현상의 편의를 위한 property
전체 길이	36	

표 62 문단의 레이아웃

4.3.5. 문단의 영역 태그

range tag 정보를 정보 수만큼 읽어 온다. range tag는 텍스트의 일정 영역을 마킹하는 용도로 사용되며, 글자 모양과는 달리 각 영역은 서로 겹칠 수 있다.(형광펜, 교정 부호 등)

Tag ID : HWPTAG_PARA_RANGE_TAG

자료형	길이(바이트)	설명
UINT32	4	영역 시작
UINT32	4	영역 끝
UINT32	4	태그(종류 + 데이터) : 상위 8비트가 종류를 하위 24비트가 종류별로 다른 설명을 부여할 수 있는 임의의 데이터를 나타낸다.
전체 길이	가변	12×n

표 63 문단의 영역 태그

4.3.6. 컨트롤 헤더

컨트롤 문자가 존재하면 컨트롤 문자로부터 존재하는 컨트롤 정보를 생성한다.

Tag ID : HWPTAG_CTRL_HEADER

자료형	길이(바이트)	설명
UINT32	4	컨트롤 ID
		컨트롤 ID 이하 속성들은 CtrlID에 따라 다르다. - 각 컨트롤 및 개체 참고
전체 길이	4	

표 64 컨트롤 헤더

4.3.7. 문단 리스트 헤더

Tag ID : HWPTAG_LIST_HEADER

자료형	길이(바이트)	설명			
INT16	2	문단 수			
UINT32	4	속성			
		범위	구분	값	설명
		bit 0~2	텍스트 방향	0	가로
				1	세로
		bit 3~4	문단의 줄바꿈	0	일반적인 줄바꿈
				1	자간을 조종하여 한 줄을 유지
				2	내용에 따라 폭이 늘어남
		bit 5~6	세로 정렬	0	top
				1	center
				2	bottom
전체 길이	6				

표 65 문단 리스트 헤더

4.3.8. 컨트롤 임의의 데이터

컨트롤의 필드 이름이나 하이퍼링크 정보를 저장한다.

Tag ID : HWPTAG_CTRL_DATA

자료형	길이(바이트)	설명
Parameter Set	가변	파라미터 셋(표 50 참조)
전체 길이	가변	

표 66 컨트롤 임의의 데이터

4.3.9. 개체 공통 속성을 포함하는 컨트롤(개체 컨트롤)

extended type의 컨트롤은 종류를 나타내는 식별 기호로 32 비트 ID가 사용된다. 컨트롤 코드가 큰 범주를 나타내는 식별 기호라고 한다면, 컨트롤 ID는 세부 분류를 나타내는 식별 기호인 셈이다.

예를 들어 단 정의 컨트롤 ID는 MAKE_4CHID('c', 'o', 'l', 'd') 같은 형식으로 정의한다.

MAKE_4CHID(a, b, c, d) (((a) << 24) | ((b) << 16) | ((c) << 8) | (d))

		컨트롤 ID	개체 공통 속성	개체 요소 속성
1	표	MAKE_4CHID('t', 'b', 'l', ' ')	✓	
2	(그리기 개체) 선 사각형 타원 호 다각형 곡선	MAKE_4CHID('\$', 'l', 'i', 'n') MAKE_4CHID('\$', 'r', 'e', 'c') MAKE_4CHID('\$', 'e', 'l', 'l') MAKE_4CHID('\$', 'a', 'r', 'c') MAKE_4CHID('\$', 'p', 'o', 'l') MAKE_4CHID('\$', 'c', 'u', 'r')	✓	✓
3	한글 97 수식	MAKE_4CHID('e', 'q', 'e', 'd')	✓	
4	그림	MAKE_4CHID('\$', 'p', 'i', 'c')	✓	✓
5	OLE	MAKE_4CHID('\$', 'o', 'l', 'e')	✓	✓
6	묶음 개체	MAKE_4CHID('\$', 'c', 'o', 'n')	✓	✓

표 67 개체 공통 속성을 포함하는 컨트롤과 컨트롤ID

자료형	길이(바이트)	설명
BYTE stream	n	개체 공통 속성(표 69 참조)
BYTE stream	n2	캡션 정보가 있으면 캡션 리스트 정보를 얻는다(표 71 참조)
전체 길이	가변	n + n2 바이트

표 68 개체 공통 속성

자료형	길이(바이트)	설명
UINT32	4	ctrl ID
UINT32	4	속성(표 70 참조)
HWPUNIT	4	세로 오프셋 값
HWPUNIT	4	가로 오프셋 값
HWPUNIT	4	width 오브젝트의 폭
HWPUNIT	4	height 오브젝트의 높이
INT32	4	z-order
HWPUNIT16 array[4]	2×4	오브젝트의 바깥 4방향 여백
UINT32	4	문서 내 각 개체에 대한 고유 아이디(instance ID)
INT32	4	쪽나눔 방지 on(1) / off(0)
WORD	2	개체 설명문 글자 길이(len)
WCHAR array[len]	2×len	개체 설명문 글자
전체 길이	가변	46 + (2×len) 바이트

표 69 개체 공통 속성

범위	구분	값	설명
bit 0	글자처럼 취급 여부		
bit 1	예약		
bit 2	줄 간격에 영향을 줄지 여부		
bit 3~4	세로 위치의 기준 (VertRelTo)	0	paper
		1	page
		2	para
bit 5~7	세로 위치의 기준에 대한 상대적인 배열 방식	0	VerRelTo이 'paper'나 'page' 이면 top, 그렇지 않으면 left
		1	VerRelTo이 'paper'나 'page' 이면 center
		2	VerRelTo이 'paper'나 'page' 이면 bottom, 그렇지 않으면 right
		3	VerRelTo이 'paper'나 'page' 이면 inside
		4	VerRelTo이 'paper'나 'page' 이면 outside

bit 8~9	가로 위치의 기준 (HorzRelTo)	0	page
		1	page
		2	column
		3	para
bit 10~12	HorzRelTo에 대한 상대적인 배열 방식		bit 5~7 참조
bit 13	VertRelTo이 'para'일 때 오브젝트의 세로 위치를 본문 영역으로 제한할지 여부	0	off
		1	on
bit 14	다른 오브젝트와 겹치는 것을 허용할지 여부		오브젝트의 위치가 본문 영역으로 제한되면 언제나 false로 간주한다.
bit 15~17	오브젝트 폭의 기준	0	paper
		1	page
		2	coloum
		3	para
		4	absolute
bit 18~19	오브젝트 높이의 기준	0	paper
		1	page
		2	absolute
bit 20	VertRelTo이 para일 때 크기 보호 여부	0	off
		1	on
bit 21~23	오브젝트 주위를 텍스트가 어떻게 흘러갈지 지정하는 옵션		
	Square	0	bound rect를 따라
	Tight	1	오브젝트의 outline을 따라
	Through	2	오브젝트 내부의 빈 공간까지
	TopAndBottom	3	좌, 우에는 텍스트를 배치하지 않음
	BehindText	4	글과 겹치게 하여 글 뒤로
	InFrontOfText	5	글과 겹치게 하여 글 앞으로
bit 24~25	오브젝트의 좌/우 어느 쪽에 글을 배치할지 지정하는 옵션	0	BothSides
		1	LeftOnly
		2	RightOnly
		3	LargestOnly
bit 26~28	이 개체가 속하는 번호 범주	0	none
		1	figure
		2	table
		3	equation

표 70 개체 공통 속성의 속성

자료형	길이(바이트)	설명
BYTE stream	n	문단 리스트 헤더(표 65 참조)
BYTE stream	12	캡션(표 67 참조)
전체 길이	가변	12+n 바이트

표 71 캡션 리스트

자료형	길이(바이트)	설명
UNIT	4	속성(표 73 참조)
HWPUNIT	4	캡션 폭(세로 방향일 때만 사용)
HWPUNIT16	2	캡션과 틀 사이 간격
HWPUNIT	4	텍스트의 최대 길이(=개체의 폭)
전체 길이	14	

표 72 캡션

범위	구분	값	설명
bit 0~1	방향	0	left
		1	right
		2	top
		3	bottom
bit 2	캡션 폭에 마진을 포함할 지 여부		가로 방향일 때만 사용

표 73 캡션 속성

4.3.9.1. 표 개체

Tag ID : HWPTAG_TABLE

자료형	길이(바이트)	설명
BYTE stream	n	개체 공통 속성(표 68 참조)
BYTE stream	n2	표 개체 속성(표 75 참조)
BYTE stream	n3	셀 리스트(표 79 참조) 셀 size×셀 개수.
전체 길이	가변	n + n2 + n3

표 74 표 개체

자료형	길이(바이트)	설명
UINT32	4	속성
UINT16	2	RowCount
UINT16	2	nCols
HWPUNIT16	2	CellSpacing
BYTE stream	8	안쪽 여백 정보(표 77 참조)
BYTE stream	2×n	Row Size
UINT16	2	Border Fill ID
UINT16	2	Valid Zone Info Size (5.0.1.0 이상)
BYTE stream	10×n	영역 속성(표 73 참조) (5.0.1.0 이상)
전체 길이	가변	22 + (2×row) + (10×zone)

표 75 표 개체 속성

범위	구분	값	설명
bit 0-1	쪽 경계에서 나눔	0	나누지 않음
		1	셀 단위로 나눔
		2	나누지 않음
bit 2	제목 줄 자동 반복 여부		

표 76 표 속성의 속성

자료형	길이(바이트)	설명
HWPUNIT16	2	왼쪽 여백
HWPUNIT16	2	오른쪽 여백
HWPUNIT16	2	위쪽 여백
HWPUNIT16	2	아래쪽 여백
전체 길이	8	

표 77 안쪽 여백 정보

자료형	길이(바이트)	설명
UINT16	2	시작 열 주소
UINT16	2	시작 행 주소
UINT16	2	끝 열 주소
UINT16	2	끝 행 주소
UINT16	2	테두리 채우기 ID
전체 길이	10	

표 78 영역 속성

자료형	길이(바이트)	설명
BYTE stream	n	문단 리스트 헤더(표 65 참조)
BYTE stream	26	셀 속성(표 80 참조)
전체 길이	가변	26+n 바이트

표 79 셀 리스트

자료형	길이(바이트)	설명
UINT16	2	셀 주소(Column, 맨 왼쪽 셀이 0부터 시작하여 1씩 증가)
UINT16	2	셀 주소(Row, 맨 위쪽 셀이 0부터 시작하여 1씩 증가)
UINT16	2	열의 병합 개수
UINT16	2	행의 병합 개수
HWPUNIT	4	셀의 폭
HWPUNIT	4	셀의 높이
HWPUNIT16 array[4]	2x4	셀 4방향 여백
UINT16	2	테두리/배경 아이디
전체 길이	26	

표 80 셀 속성

4.3.9.2. 그리기 개체(선, 사각형, 타원, 호, 다각형, 곡선)

모든 그리기 개체에 대한 serialization은 우선 base인 그리기 개체 공통 속성을 serialize한 다음 자신이 가지고 있는 개체 요소 속성을 serialize한다.

자료형	길이(바이트)	설명
BYTE stream	n	개체 요소 속성(표 82 참조)
BYTE stream	11	테두리 선 정보(표 86 참조)
BYTE stream	n2	채우기 정보(표 28 참조)
BYTE stream	12	글상자 속성이 있으면 글상자의 리스트 정보를 얻는다.
전체 길이	가변	23 + n + n2 바이트

표 81 그리기 개체 공통 속성

4.3.9.2.1. 개체 요소

Tag ID : HWPTAG_SHAPE_COMPONENT(GenShapeObject일 경우 id가 두 번 기록됨)

자료형	길이(바이트)	설명
-----	---------	----

UINT32	4	개체 컨트롤 ID
전체 길이	4	

표 82 개체 요소 속성

자료형	길이(바이트)	설명	
INT32	4	개체가 속한 그룹 내에서의 X offset	
INT32	4	개체가 속한 그룹 내에서의 Y offset	
WORD	2	몇 번이나 그룹 되었는지	
WORD	2	개체 요소의 local file version	
UINT32	4	개체 생성 시 초기 폭	
UINT32	4	개체 생성 시 초기 높이	
UINT32	4	개체의 현재 폭	
UINT32	4	개체의 현재 높이	
UINT32	4	속성	
		값	설명
		0	horz flip
1	vert flip		
HWPUNIT16	2	회전각	
INT32	4	회전 중심의 x 좌표(개체 좌표계)	
INT32	4	회전 중심의 y 좌표(개체 좌표계)	
	n	Rendering 정보(표 79 참조)	
전체 길이	가변	42+n 바이트	

표 83 개체 요소 속성

자료형	길이(바이트)	설명
WORD	2	scale matrix와 rotation matrix쌍의 개수(cnt) 초기엔 1, group할 때마다 하나씩 증가하고, ungroup할 때 마다 하나씩 감소한다.
BYTE stream	48	translation matrix(표 85 참조)
BYTE stream	cnt×48×2	scale matrix/rotation matrix sequence(표 85 참조)
전체 길이	가변	50 + (cnt×48×2) 바이트

표 84 Rendering 정보

각 matrix는 원소가 double로 표현되는 3 X 3 matrix로 구현된다. 마지막 줄(row)은 항상 0, 0, 1이기 때문에 실제 serialization에서는 마지막 줄은 빠진다. 저장되는 정보는 다음과 같다.

자료형	길이(바이트)	설명
double array[6]	8×6	3 X 2 matrix의 원소
전체 길이	48	

표 85 matrix 정보

자료형	길이(바이트)	설명
COLORREF	4	선 색상
INT16	2	선 굵기
UINT32	4	속성(표 87 참조)
BYTE	1	Outline style(표 88 참조)
전체 길이	11	

표 86 테두리 선 정보

범위	구분	값	설명	
bit 0~5	선 종류			
bit 6~9	선 끝 모양	0	round	그림일 때는 "Round", 그리기 개체들일 때는 "Flat"이 디폴트
		1	flat	
bit 10~15	화살표 시작 모양			
bit 16~21	화살표 끝 모양	0	모양 없음	
		1	화살 모양	
		2	오목한 화살모양	
		3	속이 빈 다이아몬드 모양	
		4	속이 빈 원 모양	
		5	속이 빈 사각 모양	
		6	속이 채워진 다이아몬드 모양	
		7	속이 채워진 원 모양	
		8	속이 채워진 사각 모양	
bit 22~25	화살표 시작 크기			
bit 26~29	화살표 끝 크기	0	작은-작은	
		1	작은-중간	
		2	작은-큰	
		3	중간-작은	
		4	중간-중간	
		5	중간-큰	
		6	큰-작은	
		7	큰-중간	
		8	큰-중간	
bit 30	시작부분 화살표 채움 여부			
bit 31	끝부분 화살표 채움 여부			

표 87 테두리 선 정보 속성

값	설명
0	normal
1	outer
2	inner

표 88 Outline style

자료형	길이(바이트)	설명
BYTE stream	12	그리기 개체 글상자용 텍스트 속성(표 90 참조)
BYTE stream	n	문단 리스트 헤더(표 65 참조)
전체 길이	가변	12+n 바이트

표 89 그리기 개체 글상자용 텍스트 정보

자료형	길이(바이트)	설명
HWPUNIT16	2	글상자 텍스트 왼쪽 여백
HWPUNIT16	2	글상자 텍스트 오른쪽 여백
HWPUNIT16	2	글상자 텍스트 위쪽 여백
HWPUNIT16	2	글상자 텍스트 아래쪽 여백
HWPUNIT	4	텍스트 문자열의 최대 폭 (보통 그리기 개체의 가로 크기와 동일)
전체 길이	12	

표 90 그리기 개체 글상자용 텍스트 속성

4.3.9.2.2. 선 개체

자료형	길이(바이트)	설명
BYTE stream	n	개체 공통 속성(표 68 참조)
BYTE stream	n2	그리기 개체 공통 속성(표 81 참조)
BYTE stream	18	선 개체 속성(표 92 참조)
전체 길이	가변	18 + n + n2 바이트

표 91 선 개체

Tag ID : HWPTAG_SHAPE_COMPONENT_LINE

자료형	길이(바이트)	설명
INT32	4	시작점 X 좌표
INT32	4	시작점 Y 좌표
INT32	4	끝점 X 좌표
INT32	4	끝점 Y 좌표
UINT16	2	속성. 처음 생성 시 수직 또는 수평선일 때, 선의 방향이 언제나 오른쪽(위쪽)으로 잡힘으로 인한 현상 때문에, 방향을 바로 잡아주기 위한 플래그.
전체 길이	18	

표 92 선 개체 속성

4.3.9.2.3. 사각형 개체

자료형	길이(바이트)	설명
BYTE stream	n	개체 공통 속성(표 68 참조)
BYTE stream	n2	그리기 개체 공통 속성(표 81 참조)
BYTE stream	33	사각형 개체 속성(표 94 참조)
전체 길이	가변	33 + n + n2 바이트

표 93 사각형 개체

Tag ID : HWPTAG_SHAPE_COMPONENT_RECTANGLE

자료형	길이(바이트)	설명
BYTE	1	사각형 모서리 곡률(%) 직각은 0, 둥근 모양은 20, 반원은 50, 그 외는 적당한 값을 % 단위로 사용한다.
INT32 array[4]	4×4	사각형의 좌표(x)
INT32 array[4]	4×4	사각형의 좌표(y)
전체 길이	33	

표 94 사각형 개체 속성

4.3.9.2.4. 타원 개체

자료형	길이(바이트)	설명
BYTE stream	n	개체 공통 속성(표 68 참조)
BYTE stream	n2	그리기 개체 공통 속성(표 81 참조)
BYTE stream	60	타원 개체 속성(표 96 참조)
전체 길이	가변	60 + n + n2 바이트

표 95 타원 개체

Tag ID : HWPTAG_SHAPE_COMPONENT_ELLIPSE

자료형	길이(바이트)	설명
UINT32	4	속성(표 97 참조)
INT32	4	중심 좌표의 X 값
INT32	4	중심 좌표의 Y 값
INT32	4	제1축 X 좌표 값
INT32	4	제1축 Y 좌표 값
INT32	4	제2축 X 좌표 값
INT32	4	제2축 Y 좌표 값
INT32	4	start pos x
INT32	4	start pos y
INT32	4	end pos x
INT32	4	end pos y
INT32	4	start pos x2 interval of curve(effective only when it is an arc)
INT32	4	start pos y2
INT32	4	end pos x2
INT32	4	end pos y2
전체 길이	60	

표 96 타원 개체 속성

범위	설명
bit 0	호(ARC)로 바뀌었을 때, interval을 다시 계산해야 할 필요가 있는지 여부 (interval - 원 위에 존재하는 두 점 사이의 거리)
bit 1	호(ARC)로 바뀌었는지 여부
bit 2~9	호(ARC)의 종류

표 97 타원/호 개체 속성의 속성

4.3.9.2.5. 다각형 개체

자료형	길이(바이트)	설명
BYTE stream	n	개체 공통 속성(표 68 참조)
BYTE stream	n2	그리기 개체 공통 속성(표 81 참조)
BYTE stream	n3	다각형 개체 속성(표 99 참조)
전체 길이	가변	n + n2 + n3 바이트

표 98 다각형 개체

Tag ID : HWPTAG_SHAPE_COMPONENT_POLYGON

자료형	길이(바이트)	설명
INT16	2	count of points(cnt)
INT32 array[cnt]	4×cnt	x 좌표
INT32 array[cnt]	4×cnt	y 좌표
전체 길이	가변	2 + 2(4×cnt) 바이트

표 99 다각형 개체 속성

4.3.9.2.6. 호 개체

자료형	길이(바이트)	설명
BYTE stream	n	개체 공통 속성(표 68 참조)
BYTE stream	n2	그리기 개체 공통 속성(표 81 참조)
BYTE stream	28	호 개체 속성(표 96 참조)
전체 길이	가변	28 + n + n2 바이트

표 100 호 개체

Tag ID : HWPTAG_SHAPE_COMPONENT_ARC

자료형	길이(바이트)	설명
UINT32	4	속성(표 92 참조)
INT32	4	타원의 중심 좌표 X 값
INT32	4	타원의 중심 좌표 Y 값
INT32	4	제1축 X 좌표 값
INT32	4	제1축 Y 좌표 값
INT32	4	제2축 X 좌표 값
INT32	4	제2축 Y 좌표 값
전체 길이	28	

표 101 호 개체 속성

4.3.9.2.7. 곡선 개체

자료형	길이(바이트)	설명
BYTE stream	n	개체 공통 속성(표 68 참조)
BYTE stream	n2	그리기 개체 공통 속성(표 81 참조)
BYTE stream	n3	곡선 개체 속성(표 103 참조)
전체 길이	가변	n + n2 + n3 바이트

표 102 곡선 개체

Tag ID : HWPTAG_SHAPE_COMPONENT_CURVE

자료형	길이(바이트)	설명
INT16	2	count of points(cnt)
INT32 array[cnt]	4×cnt	x 좌표
INT32 array[cnt]	4×cnt	y 좌표
BYTE array[cnt-1]	cnt-1	segment type(0 : line, 1 : curve)
전체 길이	가변	2 + 2(4×cnt) + cnt-1 바이트

표 103 곡선 개체 속성

4.3.9.3. 한글 수식 개체

자료형	길이(바이트)	설명
BYTE stream	n	개체 공통 속성(표 68 참조)
BYTE stream	n2	수식 개체 속성(표 105 참조)
전체 길이	가변	n + n2 바이트

표 104 수식 개체

Tag ID : HWPTAG_EQEDIT

자료형	길이(바이트)	설명
UINT32	4	속성. 스크립트가 차지하는 범위. 첫 비트가 켜져 있으면 줄 단위, 꺼져 있으면 글자 단위.
WORD	2	스크립트 길이(len)
WCHAR array[len]	2×len	한글 수식 스크립트(*)
HWPUNIT	4	수식 글자 크기
COLORREF	4	글자 색상
INT16	2	base line
WCHAR array[len]	2×len	수식 버전 정보
WCHAR array[len]	2×len	수식 폰트 이름
전체 길이	가변	16 + (6×len) 바이트

표 105 수식 개체 속성

*한글 수식 스크립트는 EQN 스크립트 호환이며 자세한 스펙은 뒤에 추가

4.3.9.4. 그림 개체(HWPTAG_SHAPE_COMPONENT_PICTURE)

자료형	길이(바이트)	설명
BYTE stream	n	개체 공통 속성(표 68 참조)
BYTE stream	n2	개체 요소 공통 속성(표 80 참조)
BYTE stream	n3	그림 개체 속성(표 107 참조)
전체 길이	가변	n + n2 + n3 바이트

표 106 그림 개체

자료형	길이(바이트)	설명
-----	---------	----

COLORREF	4	테두리 색
INT32	4	테두리 두께
UINT32	4	테두리 속성(표 86 참조)
INT32 array[4]	4×4	이미지의 테두리 사각형의 x 좌표(최초 그림 삽입 시 크기)
INT32 array[4]	4×4	이미지의 테두리 사각형의 y 좌표(최초 그림 삽입 시 크기)
INT32	4	자르기 한 후 사각형의 left
INT32	4	자르기 한 후 사각형의 top
INT32	4	자르기 한 후 사각형의 right
INT32	4	자르기 한 후 사각형의 bottom
BYTE stream	8	안쪽 여백 정보(표 72 참조) 표(기본값 : 141), 그림(기본값 : 0)
BYTE stream	5	그림 정보(표 27 참조)
BYTE	1	테두리 투명도
UINT32	4	문서 내 각 개체에 대한 고유 아이디(instance ID)
BYTE stream	n	그림 효과 정보
전체 길이	가변	78 + n 바이트

표 107 그림 개체 속성

자료형	길이(바이트)	설명
UINT32	4	그림 효과 정보(그림자, 네온, 부드러운 가장자리, 반사)
BYTE stream	n	각 효과 정보(표 109 ~ 표 113 참조)
전체 길이	가변	4 + n 바이트

표 108 그림 효과 속성

자료형	길이(바이트)	설명
INT32	4	그림자 스타일
float	4	그림자 투명도
float	4	그림자 흐릿하게
float	4	방향
float	4	거리
INT32	4	정렬
float	4	기울기 각도(X)
float	4	기울기 각도(Y)
float	4	확대 비율(X)
float	4	확대 비율(Y)
INT32	4	도형과 함께 그림자 회전
BYTE stream	n	그림자 색상(표 113 참조)
전체 길이	가변	44 + n 바이트

표 109 그림자 효과 속성

자료형	길이(바이트)	설명
float	4	네온 투명도
float	4	네온 반경
BYTE stream	n	네온 색상(표 113 참조)
전체 길이	가변	8 + n 바이트

표 110 네온 효과 속성

자료형	길이(바이트)	설명
float	4	부드러운 가장자리 반경
전체 길이	4	

표 111 부드러운 가장자리 효과 속성

자료형	길이(바이트)	설명
INT32	4	반사 스타일
float	4	반경
float	4	방향
float	4	거리
float	4	기울기 각도(X)
float	4	기울기 각도(Y)
float	4	확대 비율(X)
float	4	확대 비율(Y)
INT32	4	회전 스타일
float	4	시작 투명도
float	4	시작 위치
float	4	끝 투명도
float	4	끝 위치
float	4	오프셋 방향
전체 길이	53	

표 112 반사 효과 속성

자료형	길이(바이트)	설명
INT32	4	색상 타입
UINT32	4	RGB (0x00rrggbb)
UINT32	4	CMYK (0xccmmyykk)
INT32	4	스키마 타입
INT32	4	시스템 타입
INT32	4	프리셋 타입
float	4	ScR
float	4	ScG
float	4	ScB
float	4	H
float	4	S
float	4	L
UINT32	4	색상 효과 수
BYTE array[n]	8×n	색상 효과(표 111 참조)
전체 길이	가변	4 + m(4 ~ 12) + n 바이트

표 113 색상 속성

자료형	길이(바이트)	의미
INT32	4	색상 효과 종류(표 115 참조)
float	4	색상 효과 값
전체 길이	8	

표 114 색상 효과 속성

값	설명

0	alpha
1	alpha_mod
2	alpha_off
3	red
4	red_mod
5	red_off
6	green
7	green_mod
8	green_off
9	blue
10	blue_mod
11	blue_off
12	hue
13	hue_mod
14	hue_off
15	sat
16	sat_mod
17	sat_off
18	lum
19	lum_mod
20	lum_off
21	shade
22	tint
23	gray
24	comp
25	gamma
26	inv_gamma
27	inv

표 115 색상 효과 종류

자료형	길이(바이트)	설명
HWPUNIT	4X2	그림 최초 생성 시 기준 이미지 크기
INT8	1	이미지 투명도
전체 길이	9	

표 116 그림 추가 속성

4.3.9.5. OLE 개체(HWPTAG_SHAPE_COMPONENT_OLE)

자료형	길이(바이트)	설명
BYTE stream	n	개체 공통 속성(표 68 참조)
BYTE stream	24	OLE 개체 속성(표 118 참조)
전체 길이	가변	24 + n 바이트

표 117 OLE 개체

자료형	길이(바이트)	설명
UINT16	2	속성(표 119 참조)
INT32	4	오브젝트 자체의 extent x크기
INT32	4	오브젝트 자체의 extent y크기
UINT16	2	오브젝트가 사용하는 스토리지의 BinData ID

COLORREF	4	테두리 색
INT32	4	테두리 두께
UINT32	4	테두리 속성(표 87 참조)
전체 길이	24	

표 118 OLE 개체 속성

범위	구분	값	설명
bit 0~7			DVASPECT_CONTENT = 1, DVASPECT_THUMBNAIL = 2, DVASPECT_ICON = 4, DVASPECT_DOCPRINT = 8
bit 8			TRUE if moniker is assigned
bit 9~15			베이스라인. 0은 디폴트(85%)를 뜻하고, 1~101이 0~100%를 나타낸다. 현재는 수식만이 베이스라인을 별도로 가진다.
bit 16~21	개체 종류	0	Unknown
		1	Embedded
		2	Link
		3	Static
		4	Equation

표 119 OLE 개체 속성의 속성

bit 0-7에 대한 자세한 설명은 MSDN을 참고
MFC COleClientItem::m_nDrawAspect

bit 8에 대한 자세한 설명은 MSDN을 참고
MFC COleClientItem::m_bMoniker

4.3.9.6. 차트 개체

차트는 본문에 Ole 개체로 저장되며, 차트 내용은 Ole Compound 파일로 저장된다.

차트 파일의 최상위 storage에 "Contents", "OOXMLChartContents" stream이 저장될 수 있다.

기존 hwp(한글 Neo) 차트에서는 "Contents" stream만 존재한다.

한글 2018 한글에서는 OOXML 차트를 저장할 수 있고, "OOXMLChartContents"와 "Contents" stream에 차트를 저장할 수 있다.

한글 2018에서 차트에서 "Contents" stream은 OOXML 차트를 구 버전(한글 Neo)의 차트로 변환한 예비 데이터이다.

"Contents" stream의 내용은 첨부 파일("한글문서파일형식_차트_revision1.2_원본.hwp")의 내용을 참조하면 된다.

"OOXMLChartContents" stream은 MS OOXML에서 차트를 저장하는 방식으로, 최상위 element는 "chartSpace" element이며, 내용은 ISO/IEC 29500나 ECMA 376의 DrawingML의 Chart 부분을 참조하면 된다.

4.3.9.7. 묶음 개체(HWPTAG_SHAPE_COMPONENT_CONTAINER)

자료형	길이(바이트)	설명
BYTE stream	n	개체 공통 속성(표 68 참조)
BYTE stream	n2	묶음 개체 속성(표 121 참조)
BYTE stream	n3	개체 속성 x 묶음 개체의 갯수. (묶음 가능 개체 : 그리기 개체, OLE, 그림, 묶음 개체)
전체 길이	가변	n + n2 + n3 바이트

표 120 묶음 개체

자료형	길이(바이트)	설명
WORD	2	개체의 개수(n)
UINT32 array[n]	4×n	개체의 컨트롤 ID array
전체 길이	가변	2 + (4×n) 바이트

표 121 묶음 개체 속성

4.3.9.8. 동영상 개체(HWPTAG_VIDEO_TDATA)

자료형	길이(바이트)	설명
BYTE stream	n	개체 공통 속성(표 68 참조)
BYTE stream	n2	동영상 개체 속성
전체 길이	가변	n + n2 바이트

표 122 동영상 개체

자료형	길이(바이트)	설명
INT32	4	동영상 타입(표 124 참조)
BYTE stream	n	동영상 타입에 따라 길이가 다름(표 125, 표 126 참조)
전체 길이	가변	n + n2 바이트

표 123 동영상 개체 속성

값	설명
0	로컬 동영상
1	웹 동영상

표 124 동영상 타입

자료형	길이(바이트)	설명
UINT16	2	비디오 파일의 사용하는 스토리지의 BinData ID
UINT16	2	썸네일 파일이 사용하는 스토리지의 BinData ID
전체 길이	4	

표 125 로컬 동영상 속성

자료형	길이(바이트)	설명
-----	---------	----

WCHAR array[len]	2xlen	웹 태그
UINT16	2	썸네일 파일이 사용하는 스토리지의 BinData ID
전체 길이	(2xlen)+2	

표 126 웹 동영상 속성

4.3.10. 개체 이외의 컨트롤

	의미	컨트롤 ID	문단 리스트	파일 태그
1	구역 정의	MAKE_4CHID('s', 'e', 'c', 'd')	✓	HWPTAG_CTRL_HEAD로 부터 시작하며 ctrlid로 각 개체를 확인 할 수 있다.
2	단 정의	MAKE_4CHID('c', 'o', 'l', 'd')		
3	머리말 / 꼬리말	MAKE_4CHID('h', 'e', 'a', 'd') / MAKE_4CHID('f', 'o', 'o', 't')	✓	
4	각주 / 미주	MAKE_4CHID('f', 'n', 't', 'y') / MAKE_4CHID('e', 'n', 't', 'y')	✓	
5	자동번호	MAKE_4CHID('a', 't', 'n', 'o')		
6	새 번호 지정	MAKE_4CHID('n', 'w', 'n', 'o')		
7	감추기	MAKE_4CHID('p', 'g', 'h', 'd')		
8	홀/짝수 조정	MAKE_4CHID('p', 'g', 'c', 't')		
9	쪽 번호 위치	MAKE_4CHID('p', 'g', 'n', 'p')		
10	찾아보기 표식	MAKE_4CHID('i', 'd', 'x', 'm')		
11	책갈피	MAKE_4CHID('b', 'o', 'k', 'm')		
12	글자 겹침	MAKE_4CHID('t', 'c', 'p', 's')		
13	덧말	MAKE_4CHID('t', 'd', 'u', 't')		
14	숨은 설명	MAKE_4CHID('t', 'c', 'm', 't')	✓	
15	필드 시작	필드 컨트롤 ID		

표 127 개체 이외의 컨트롤과 컨트롤 ID

특정 컨트롤은 정보 이외의 문단 리스트를 가질 수 있다.

	컨트롤 ID
FIELD_UNKNOWNN	MAKE_4CHID('%', 'u', 'n', 'k')
FIELD_DATE	MAKE_4CHID('%', 'd', 't', 'e')
FIELD_DOCDATE	MAKE_4CHID('%', 'd', 'd', 't')
FIELD_PATH	MAKE_4CHID('%', 'p', 'a', 't')
FIELD_BOOKMARK	MAKE_4CHID('%', 'b', 'm', 'k')
FIELD_MAILMERGE	MAKE_4CHID('%', 'm', 'm', 'g')
FIELD_CROSSREF	MAKE_4CHID('%', 'x', 'r', 'f')
FIELD_FORMULA	MAKE_4CHID('%', 'f', 'm', 'u')
FIELD_CLICKHERE	MAKE_4CHID('%', 'c', 'l', 'k')
FIELD_SUMMARY	MAKE_4CHID('%', 's', 'm', 'r')
FIELD_USERINFO	MAKE_4CHID('%', 'u', 's', 'r')
FIELD_HYPERLINK	MAKE_4CHID('%', 'h', 'l', 'k')
FIELD_REVISION_SIGN	MAKE_4CHID('%', 's', 'i', 'g')
FIELD_REVISION_DELETE	MAKE_4CHID('%', '%', '*', 'd')
FIELD_REVISION_ATTACH	MAKE_4CHID('%', '%', '*', 'a')
FIELD_REVISION_CLIPPING	MAKE_4CHID('%', '%', '*', 'C')
FIELD_REVISION_SAWTOOTH	MAKE_4CHID('%', '%', '*', 'S')
FIELD_REVISION_THINKING	MAKE_4CHID('%', '%', '*', 'T')
FIELD_REVISION_PRAISE	MAKE_4CHID('%', '%', '*', 'P')
FIELD_REVISION_LINE	MAKE_4CHID('%', '%', '*', 'L')
FIELD_REVISION_SIMPLECHANGE	MAKE_4CHID('%', '%', '*', 'c')
FIELD_REVISION_HYPERLINK	MAKE_4CHID('%', '%', '*', 'h')

FIELD_REVISION_LINEATTACH	MAKE_4CHID('%', '%', '*', 'A')
FIELD_REVISION_LINELINK	MAKE_4CHID('%', '%', '*', 'i')
FIELD_REVISION_LINETRANSFER	MAKE_4CHID('%', '%', '*', 't')
FIELD_REVISION_RIGHTMOVE	MAKE_4CHID('%', '%', '*', 'r')
FIELD_REVISION_LEFTMOVE	MAKE_4CHID('%', '%', '*', 'l')
FIELD_REVISION_TRANSFER	MAKE_4CHID('%', '%', '*', 'n')
FIELD_REVISION_SIMPLEINSERT	MAKE_4CHID('%', '%', '*', 'e')
FIELD_REVISION_SPLIT	MAKE_4CHID('%', 's', 'p', 'l')
FIELD_REVISION_CHANGE	MAKE_4CHID('%', '%', 'm', 'r')
FIELD_MEMO	MAKE_4CHID('%', '%', 'm', 'e')
FIELD_PRIVATE_INFO_SECURITY	MAKE_4CHID('%', 'c', 'p', 'r')
FIELD_TABLEOFCONTENTS	MAKE_4CHID('%', 't', 'o', 'c')

표 128 필드 컨트롤 ID

4.3.10.1. 구역 정의

자료형	길이(바이트)	설명
UINT32	4	속성(표 130 참조)
HWPUNIT16	2	동일한 페이지에서 서로 다른 단 사이의 간격
HWPUNIT16	2	세로로 줄맞춤을 할지 여부 0 = off, 1 - n = 간격을 HWPUNIT 단위로 지정
HWPUNIT16	2	가로로 줄맞춤을 할지 여부 0 = off, 1 - n = 간격을 HWPUNIT 단위로 지정
HWPUNIT	4	기본 탭 간격(hwpunit 또는 relative characters)
UINT16	2	번호 문단 모양 ID
UINT16	2	쪽 번호 (0 = 앞 구역에 이어, n = 임의의 번호로 시작)
UINT16 array[3]	2×3	그림, 표, 수식 번호 (0 = 앞 구역에 이어, n = 임의의 번호로 시작)
UINT16	2	대표Language(Language값이 없으면(==0), Application에 지정된 Language) 5.0.1.5 이상
전체 길이	26	
하위 레코드 자료형	길이(바이트)	설명
BYTE stream	40	용지설정 정보(표 131 참조)
BYTE stream	26	각주 모양 정보(표 133 참조)
BYTE stream	26	미주 모양 정보(표 133 참조)
BYTE stream	12	쪽 테두리/배경 정보(표 135 참조)
BYTE stream	10	양 쪽, 홀수 쪽, 짝수 쪽의 바탕쪽 내용이 있으면 바탕쪽 정보를 얻는다. 바탕쪽 정보는 문단 리스트를 포함한다(표 137 참조)
전체 길이	140	

표 129 구역 정의

범위	설명
bit 0	머리말을 감출지 여부
bit 1	꼬리말을 감출지 여부
bit 2	바탕쪽을 감출지 여부
bit 3	테두리를 감출지 여부
bit 4	배경을 감출지 여부
bit 5	쪽 번호 위치를 감출지 여부
bit 8	구역의 첫 쪽에만 테두리 표시 여부
bit 9	구역의 첫 쪽에만 배경 표시 여부
bit 16 ~ 18	텍스트 방향(0 : 가로 1 : 세로)

bit 19	빈 줄 감춤 여부
bit 20~21	구역 나눔으로 새 페이지가 생길 때의 페이지 번호 적용할지 여부
bit 22	원고지 정서법 적용 여부

표 130 구역 정의 속성

4.3.10.1.1. 용지 설정

Tag ID : HWPTAG_PAGE_DEF

자료형	길이(바이트)	설명
HWPUNIT	4	용지 가로 크기
HWPUNIT	4	용지 세로 크기
HWPUNIT	4	용지 왼쪽 여백
HWPUNIT	4	오른쪽 여백
HWPUNIT	4	위 여백
HWPUNIT	4	아래 여백
HWPUNIT	4	머리말 여백
HWPUNIT	4	꼬리말 여백
HWPUNIT	4	제본 여백
UINT32	4	속성(표 132 참조)
전체 길이	40	

표 131 용지 설정

범위	구분	값	설명
bit 0	용지 방향	0	좁게
		1	넓게
bit 1~2	제책 방법	0	한쪽 편집
		1	맞쪽 편집
		2	위로 넘기기

표 132 용지 설정 속성

4.3.10.1.2. 각주/미주 모양

Tag ID : HWPTAG_FOOTNOTE_SHAPE

자료형	길이(바이트)	설명
UINT32	4	속성(표 134 참조)
WCHAR	2	사용자 기호
WCHAR	2	앞 장식 문자
WCHAR	2	뒤 장식 문자
UINT16	2	시작 번호
HWPUNIT16	2	구분선 길이
HWPUNIT16	2	구분선 위 여백
HWPUNIT16	2	구분선 아래 여백
HWPUNIT16	2	주석 사이 여백
UINT8	1	구분선 종류(테두리/배경의 테두리 선 종류 참조)
UINT8	1	구분선 굵기(테두리/배경의 테두리 선 굵기 참조)
CORORREF	4	구분선 색상(테두리/배경의 테두리 선 색상 참조)
전체 길이	26	

표 133 각주/미주 모양

범위	구분	값	설명
bit 0~7	번호 모양. 0~16 은 범용. 0x80, 0x81은 각주/미주 전용	0	1, 2, 3
		1	동그라미 쳐진 1, 2, 3
		2	I, II, III
		3	i, ii, iii
		4	A, B, C
		5	a, b, c
		6	동그라미 쳐진 A, B, C
		7	동그라미 쳐진 a, b, c
		8	가, 나, 다
		9	동그라미 쳐진 가,나,다
		10	ㄱ, ㄴ, ㄷ
		11	동그라미 쳐진 ㄱ,ㄴ,ㄷ
		12	일, 이, 삼
		13	一, 二, 三
		14	동그라미 쳐진 一,二,三
		15	갑, 을, 병, 정, 무, 기, 경, 신, 임, 계
		16	甲, 乙, 丙, 丁, 戊, 己, 庚, 辛, 壬, 癸
	0x80	4가지 문자가 차례로 반복	
	0x81	사용자 지정 문자 반복	
bit 8~9	한 페이지 내에서 각주를 다단에 위치시킬 방법	0	(각주인 경우) 각 단마다 따로 배열
		1	(각주인 경우) 통단으로 배열
		2	(각주인 경우) 가장 오른쪽 단에 배열
		0	(미주인 경우) 문서의 마지막
		1	(미주인 경우) 구역의 마지막
bit 10~11	numbering	0	앞 구역에 이어서
		1	현재 구역부터 새로 시작
		2	쪽마다 새로 시작(각주 전용)
bit 12	각주 내용 중 번호 코드의 모양을 위 첨자 형식으로 할지 여부		
bit 13	텍스트에 이어 바로 출력할지 여부		

표 134 각주/미주 모양 속성

4.3.10.1.3. 쪽 테두리/배경

Tag ID : HWPTAG_PAGE_BORDER_FILL

자료형	길이(바이트)	설명
UINT	4	속성(표 136 참조)
HWPUNIT16	2	테두리/배경 위치 왼쪽 간격
HWPUNIT16	2	테두리/배경 위치 오른쪽 간격
HWPUNIT16	2	테두리/배경 위치 위쪽 간격
HWPUNIT16	2	테두리/배경 위치 아래쪽 간격
UINT16	2	테두리/배경 ID
전체 길이	12	

표 135 쪽 테두리/배경

범위	구분	값	설명
bit 0	위치 기준	0	본문 기준
		1	종이 기준
bit 1	머리말 포함	0	미포함
		1	포함
bit 2	꼬리말 포함	0	미포함
		1	포함
bit 3 ~ 4	채울 영역	0	종이
		1	쪽
		2	테두리

표 136 쪽 테두리/배경 속성

자료형	길이(바이트)	설명
HWPUNIT	4	텍스트 영역의 폭
HWPUNIT	4	텍스트 영역의 높이
BYTE	1	각 비트가 해당 레벨의 텍스트에 대한 참조를 했는지 여부
BYTE	1	각 비트가 해당 레벨의 번호에 대한 참조를 했는지 여부
전체 길이	10	

표 137 바탕쪽 정보

4.3.10.2. 단 정의

자료형	길이(바이트)	설명
UINT16	2	속성의 bit 0-15(표 138 참조)
HWPUNIT16	2	단 사이 간격
WORD array[cnt]	2×cnt	단 너비가 동일하지 않으면, 단의 개수만큼 단의 폭
UINT16	2	속성의 bit 16-32(표 139 참조)
UINT8	1	단 구분선 종류(테두리/배경의 테두리 선 종류 참조)
UINT8	1	단 구분선 굵기(테두리/배경의 테두리 선 굵기 참조)
CORORREF	4	단 구분선 색상(테두리/배경의 테두리 선 색상 참조)
전체 길이	가변	12 + (2×cnt) 바이트

표 138 단 정의

4.3.10.3. 머리말/꼬리말

범위	구분	값	설명
bit 0~1	단 종류	0	일반 단단
		1	배분 단단
		2	평행 단단
bit 2~9	단 개수(cnt)	1-255	
bit 10~11	단 방향 지정	0	왼쪽부터
		1	오른쪽부터
		2	맞쪽
bit 12	단 너비 동일하게 여부		

표 139 단 정의 속성

문단 리스트를 포함한다.

자료형	길이(바이트)	설명
UINT32	4	속성(표 141 참조)
HWPUNIT	4	텍스트 영역의 폭
HWPUNIT	4	텍스트 영역의 높이
BYTE	1	각 비트가 해당 레벨의 텍스트에 대한 참조를 했는지 여부
BYTE	1	각 비트가 해당 레벨의 번호에 대한 참조를 했는지 여부
전체 길이	14	

표 140 머리말/꼬리말

범위	구분	값	설명
bit 0~1	머리말이 적용될 범위(페이지 종류)	0	양 쪽
		1	짝수 쪽만
		2	홀수 쪽만

표 141 머리말/꼬리말 속성

4.3.10.4. 각주/미주

각주/미주는 문단 리스트 외에 속성을 갖지 않는다.

하지만 쓰레기 값이나 불필요한 업데이트를 줄이기 위해 8 byte를 serialize한다.

4.3.10.5. 자동 번호

자료형	길이(바이트)	설명
UINT32	4	속성(표 143 참조)
UINT16	2	번호
WCHAR	2	사용자 기호
WCHAR	2	앞 장식 문자
WCHAR	2	뒤 장식 문자
전체 길이	12	

표 142 자동 번호

범위	구분	값	설명
bit 0~3	번호 종류	0	쪽 번호
		1	각주 번호
		2	미주 번호
		3	그림 번호
		4	표 번호
		5	수식 번호
bit 4~11	번호 모양		표 134 참조
bit 12	superscript		각주에서만 사용된다. 각주 내용 중 번호 코드의 모양을 위 첨자 형식으로 할지 여부.

표 143 자동 번호 속성

4.3.10.6. 새 번호 지정

자료형	길이(바이트)	설명
UINT32	4	속성
		범위
		bit 0~3
UINT16	2	번호
전체 길이	8	

표 144 새 번호 지정

4.3.10.7. 감추기

자료형	길이(바이트)	설명		
UINT	2	속성		
		구분	값	설명
		감출 대상	0x00000001	머리말
			0x00000002	꼬리말
			0x00000004	바탕쪽
			0x00000008	테두리
			0x00000010	배경
0x00000020	쪽 번호 위치			
전체 길이	2			

표 145 감추기

4.3.10.8. 흘/짝수 조정

자료형	길이(바이트)	설명			
UINT32	4	속성			
		범위	구분	값	설명
		bit 0~1	흘/짝수 구분	0	양 쪽
				1	짝수 쪽
2	홀수 쪽				
전체 길이	4				

표 146 흘/짝수 조정

4.3.10.9. 쪽 번호 위치

자료형	길이(바이트)	설명
UINT32	4	속성(표 148 참조)
WCHAR	2	사용자 기호
WCHAR	2	앞 장식 문자
WCHAR	2	뒤 장식 문자
WCHAR	2	항상 "-"
전체 길이	12	

표 147 쪽 번호 위치

범위	구분	값	설명
bit 0~7	번호 모양		표 143 참조
bit 8~11	번호의 표시 위치	0	쪽 번호 없음
		1	왼쪽 위
		2	가운데 위
		3	오른쪽 위
		4	왼쪽 아래
		5	가운데 아래
		6	오른쪽 아래
		7	바깥쪽 위
		8	바깥쪽 아래
		9	안쪽 위
10	안쪽 아래		

표 148 쪽 번호 위치 속성

4.3.10.10. 찾아보기 표식

자료형	길이(바이트)	설명
WORD	2	키워드 길이(len1)
WCHAR array[len1]	2×len1	찾아보기에 사용할 첫 번째 키워드
WORD	2	키워드 길이(len2)
WCHAR array[len2]	2×len2	찾아보기에 사용할 두 번째 키워드
UINT16	2	dummy
전체 길이	가변	6 + (2×len1) + (2×len2) 바이트

표 149 찾아보기 표식

4.3.10.11. 책갈피

책갈피 컨트롤은 책갈피로서 갖는 정보로서 '책갈피이름'밖에 없으며, 컨트롤 임의의 데이터인 HWPTAG_CTRL_DATA로 레코드된다. (Tag: HWPTAG_CTRL_DATA 참조)

4.3.10.12. 글자 겹침

자료형	길이(바이트)	설명
UINT32	4	ctrl ID
WORD	2	겹칠 글자 길이(len)
WCHAR array[len]	2×len	겹칠 글자
UINT8	1	테두리 타입
INT8	1	내부 글자 크기
UINT8	1	테두리 내부 글자 펼침
UINT8	1	테두리 내부 글자 속성 아이디의 수 (cnt)
UINT array[cnt]	4 x cnt	테두리 내부 글자의 charshapeid 의 배열
전체 길이	가변	10 + (2×len) + (4×cnt)

표 150 글자 겹침

4.3.10.13. 덧말

자료형	길이(바이트)	설명			
WORD	2	main Text 길이			
WCHAR	2	main Text			
WORD	2	sub Text 길이			
WCHAR	2	sub Text			
UINT	4	구분	값	설명	
				0	위
				1	아래
2	가운데				
UINT	4	Fsize ratio			
UINT	4	Option			
UINT	4	Style number			
UINT	4	구분	값	설명	
				0	양쪽 정렬
				1	왼쪽 정렬
				2	오른쪽 정렬
				3	가운데 정렬
				4	배분 정렬
5	나눔 정렬(공백에만 배분)				
전체 길이	18				

표 151 덧말

4.3.10.14. 숨은 설명

문단 리스트만을 포함한다.(문서 보안 레벨에 따라 숨은 설명 데이터들은 무효화될 수 있다.)

4.3.10.15. 필드 시작

자료형	길이(바이트)	설명
UINT32	4	ctrl ID
UINT	4	속성(표 153 참조)
BYTE	1	기타 속성
WORD	2	command 길이(len)
WCHAR array[len]	2×len	command(각 필드 종류마다 처리해야할 고유 정보)
UINT32	4	id(문서 내 고유 아이디)
전체 길이	가변	15 + (2×len)

표 152 필드

범위	구분	값	설명
bit 0	읽기 전용 상태에서도 수정 가능한지 여부		
bit 11 ~ 14	하이퍼링크 필드 업데이트 시 글자 속성 업데이트 종류	0x1	열어보지 않은 링크
		0x2	열어본 링크
		0x4	링크 생성
bit 15	필드 내용이 수정되었는지 여부		

표 153 필드 속성

4.4. 문서 이력 관리

문서 이력 관리에서 사용되는 데이터 레코드는 다음과 같다.

4.4.1. 문서 이력 관리란

한글 '문서 이력 관리 정보'는 한글 2005(6.5.0.724), 문서 형식 버전(Doc 5.0.1.7)부터 지원한다.

한글 메뉴의 "파일-문서 이력 관리"에서 표시 및 생성되는 문서의 이력 정보를 저장하는 장소이다.(한글 2005에서 한글 2007까지는 '버전 비교(파일-버전 비교)'라는 이름으로 '문서 이력 관리 정보' 기능을 제공하였다.)

문서 이력 정보의 각각의 아이템은 "히스토리" 혹은 "히스토리 아이템"이라 하며 한글 Compound 구조 내에서 각 아이템은 "DocHistory"라는 스토리지 내부에 VersionLog%d(%d는 버전) 이름의 스트림으로 저장된다. 또한, 각각 아이템은 압축, 암호화되어 저장된다.

이력 정보 데이터를 "DocHistory"라는 새로운 스토리지로 저장한다.

4.4.2. 문서 이력 관리 레코드 정보

4.4.2.1. 히스토리 아이템 정보 시작

Tag ID : HISTORY_RECORD_TYPE_STAG (0x10)

내 용	첨부 데이터 길이 및 Type	
히스토리 아이템 정보 시작	WORD	flag
	UINT	option

표 154 히스토리 아이템 정보 시작

- ✓ flag : 각 아이템에 대한 포함 레코드 flag
 - HISTORY_INFO_FLAG_VERSION (0x01) : 버전 존재
 - HISTORY_INFO_FLAG_DATE (0x02) : 날짜 존재
 - HISTORY_INFO_FLAG_WRITER (0x04) : 작성자 존재
 - HISTORY_INFO_FLAG_DESCRIPTION (0x08) : 설명 존재
 - HISTORY_INFO_FLAG_DIFFDATA (0x10) : Diff Data 존재
 - HISTORY_INFO_FLAG_LASTDOCDATA : 최근 문서 존재 (기록하지 않음, 필수)
 - HISTORY_INFO_FLAG_LOCK (0x40) : 현재 히스토리 아이템 Lock 상태
- ✓ option : 버전 정보 관련 공통 옵션
 - HWPVERSION_AUTOSAVE (0x00000001) : 문서 저장 시 자동 저장

4.4.2.2. 히스토리 아이템 정보 끝

Tag ID : HISTORY_RECORD_TYPE_ETAG (0x11)

내 용	첨부 데이터 Type
히스토리 아이템 정보 끝	NONE

표 155 히스토리 아이템 정보 끝

4.4.2.3. 히스토리 아이템 버전

Tag ID : HISTORY_RECORD_TYPE_VERSION (0x20)

내 용	첨부 데이터 Type
히스토리 아이템 버전	DWORD

표 156 히스토리 아이템 버전

4.4.2.4. 히스토리 날짜

Tag ID : HISTORY_RECORD_TYPE_DATE (0x21)

내 용	첨부 데이터 Type
히스토리 날짜	SYSTEMDATE

표 157 히스토리 날짜

4.4.2.5. 히스토리 작성자

Tag ID : HISTORY_RECORD_TYPE_WRITER (0x22)

내 용	첨부 데이터 Type
히스토리 작성자	WCHAR

표 158 히스토리 작성자

4.4.2.6. 히스토리 설명

Tag ID : HISTORY_RECORD_TYPE_DESCRIPTION (0x23)

내 용	첨부 데이터 Type
히스토리 설명	WCHAR

표 159 히스토리 설명

4.4.2.7. 비교 정보

Tag ID : HISTORY_RECORD_TYPE_DIFFDATA (0x30)

내 용	첨부 데이터 Type
비교 정보 : DiffML	WCHAR

표 160 비교 정보

4.4.2.8. 가장 마지막 최근 문서

Tag ID : HISTORY_RECORD_TYPE_LASTDOCDATA (0x31)

내 용	첨부 데이터 Type
가장 마지막 최근 문서 (HWPML)	WCHAR

표 161 가장 마지막 최근 문서

히스토리 아이템 저장 시 시작은 HISTORY_RECORD_TYPE_STAG 레코드로 시작하며, 아이템 내용의 끝은 HISTORY_RECORD_TYPE_ETAG 레코드로 종료한다.

변경 사항 이력

- **revision 1.3:20181108**
 - 참고 문헌
 - 차트 개체
 - 동영상 개체
 - 그림 추가 속성
 - 파일 인식 정보 추가
 - 수식 개체 속성 추가
 - 글머리표 추가
 - 필드 컨트롤 ID 추가
 - 개체 공통 속성 수정
 - 문단번호 수정
 - 내용 중 일부 오타 수정

- **revision 1.2:20141009**
 - 한글 문서 파일 구조 파트별로 구성
 - 5.0 일부 상세 내용 추가
 - 내용 중 일부 오타 수정
 - 3.0, HWPML 제외

- **revision 1.1:20110124**
 - 저작권 내용 수정
 - 내용 중 일부 오타 수정

- **revision 1.0:20100701**
 - 한글 문서 파일 형식 공개

한글 문서 파일 구조 5.0

발행처 (주) 한글과컴퓨터

주 소 (우) 463-400

경기도 성남시 분당구 대왕판교로

644번길 49 한컴타워 10층

전화: (031) 627-7000

팩스: (031) 627-7709
