In Kyoto, WG4 agreed that this annex should focus on additional requirements imposed by OPC. John and Chris probably created a rewirte, but it was lost unfortunately. Murata created this rewrite on the basis of the draft before the Kyoto meeting.

There are significant changes:

* + 1. Requirements on producers are reworded as requirements on ZIP archives
		2. “must” is rewritten as “shall”
		3. “Yes” is replaced by Blank since no additional requirements exist
		4. The field “version needed to extract” is required to be 4.5. No other values are allowed.

Chris wrote:

One piece of information I think is lost in my current draft is the records for which “supported” is “no” but “pass-through on editing” is set to “yes” (this includes the one DR 10-0048 was originally directed at). I’m wondering whether we could introduce a setting of “Ignored”, meaning that the record is permitted to exist in an OPC file but an OPC consumer must not use its content. It would then be down to implementations to determine whether to round-trip the record or not.

The current draft simply disallows such records and do not impose any requirements on error recovery by consumers.

Tables reworked using the following tenets:

* OPC should state what is allowed in packages, not implementations
* Supported on comsuption / supported on production should be consolidated

Annex C.
(normative)
ZIP Appnote.txt Clarifications

The ZIP specification imposes requirements on ZIP archives and those on applications that handle ZIP archives.

This appendix shows additional requirements on ZIP archives. A ZIP archive representing an abstract package shall satisfy these requirements.

This appendix also shows additional requirements on applications that handle ZIP packages. A consumer of ZIP packages shall satisfy these requirements..

C.1 Archive File Header Consistency

Note: Data describing files stored in the archive are substantially duplicated in the Local File Headers and Data Descriptors, and in the File headers within the Central Directory Record.

The ZIP archive shall hold equal values in the appropriate fields of every File Header within the Central Directory and the corresponding Local File Header and Data Descriptor pair, when the Data Descriptor exists, except as described in Table C–5 for bit 3 of general-purpose bit flags.

C.2 Data Descriptor Signature

Package consumers shall read packages, whether or not a 4-byte signature value 0x08074b50 exists at the beginning of Data Descriptors, immediately before the crc-32 field.

C.3 Table Key

* Blank — No additional requirements
* “No” — a ZIP archive shall not contain this record or field.
* “Partially” — A “Partially” value for a record indicates that additional requirements on this record or field are defined elsewhere.

Table C–1,“Support for records”, specifies the additional requirements in regard to particular top-level records or fields described in the ZIP Appnote.txt.

Table C–1. Support for records

|  |  |  |  |
| --- | --- | --- | --- |
| Record name  | Additional requirement on packages | Additional requirements on consumers |  |
| Local File Header | Partially — details below |  |  |
| File data |  |  |  |
| Data descriptor  |  |  |  |
| Archive decryption header | No |  |  |
| Archive extra data record | No |  |  |
| Central directory structure: File header | Partially — details below |  |  |
| Central directory structure: Digital signature |   | Shall ignore the signature data |  |
| Zip64 end of central directory record V1 (from spec version 4.5) | Partially — details below |  |  |
| Zip64 end of central directory record V2 (from spec version 6.2) | No |  |  |
| Zip64 end of central directory locator | Partially — details below |  |  |
| End of central directory record | Partially — details below |  |  |

Table C–2, “Support for record components”, specifies the additional requirements in regard to individual record components described in the ZIP Appnote.txt.

Table C–2. Support for record components

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Record | Field | Additional requirement on packages | Additional requirements on consumers |  |
| Local File Header | Local file header signature |  |  |  |
| Version needed to extract  | Partially — see Table C–3 |  |  |
| General purpose bit flag | Partially — see Table C–5 |  |  |
| Compression method  | Partially — see Table C–4 |  |  |
| Last mod file time |  |  |  |
| Last mod file date |  |  |  |
| Crc-32 |  |  |  |
| Compressed size |  |  |  |
| Uncompressed size |  |  |  |
| File name length |  |  |  |
| Extra field length |  |  |  |
| File name (variable size) |  |  |  |
| Extra field (variable size) | Partially — see Table C–6 |  |  |
| Central directory structure: File header | Central file header signature |  |  |  |
| version made by: high byte  |  |  |  |
| Version made by: low byte |  |  |  |
| Version needed to extract  | Partially — see Table C–3 |  |  |
| General purpose bit flag | Partially — see Table C–5 |  |  |
| Compression method | Partially — see Table C–4 |  |  |
| Last mod file time |  |  |  |
| Last mod file date |  |  |  |
| Crc-32  |  |  |  |
| Compressed size |  |  |  |
| Uncompressed size |  |  |  |
| File name length |  |  |  |
| Extra field length |  |  |  |
| File comment length |  |  |  |
| Disk number start | Partially — no multi disk archives |  |  |
| Internal file attributes |  |  |  |
| External file attributes  |  |  |  |
| Relative offset of local header  |  |  |  |
| File name (variable size) |  |  |  |
| Extra field (variable size) | Partially — see Table C–6 |  |  |
| File comment (variable size) |  |  |  |
| Zip64 end of central directory V1 (from spec version 4.5, only used when needed) | Zip64 end of central directory signature |  |  |  |
| Size of zip64 end of central directory  |  |  |  |
| Version made by: high byte |  |  |  |
| Version made by: low byte |  |  |  |
| Version needed to extract (see Table C–3 for details) | Partially – the value shall be 4.5. |  |  |
| Number of this disk | Partially — no multi disk archives |  |  |
| Number of the disk with the start of the central directory  | Partially — no multi disk archives |  |  |
| Total number of entries in the central directory on this disk  |  |  |  |
| Total number of entries in the central directory |  |  |  |
| Size of the central directory |  |  |  |
| Offset of start of central directory with respect to the starting disk number |  |  |  |
| Zip64 extensible data sector  |  |  |  |
| Zip64 end of central directory locator (only used when needed) | Zip64 end of central dir locator signature |  |  |  |
| Number of the disk with the start of the zip64 end of central directory | Partially — no multi disk archives |  |  |
| Relative offset of the zip64 end of central directory record |  |  |  |
| Total number of disks | Partially — no multi disk archives |  |  |
| End of central directory record | End of central dir signature |  |  |  |
| Number of this disk | Partially — no multi disk archives |  |  |
| Number of the disk with the start of the central directory | Partially — no multi disk archive |  |  |
| Total number of entries in the central directory on this disk |  |  |  |
| Total number of entries in the central directory |  |  |  |
| Size of the central directory |  |  |  |
| Offset of start of central directory with respect to the starting disk number |  |  |  |
| ZIP file comment length |  |  |  |
| ZIP file comment |  |  |  |

Table C–3, “Support for Version Needed to Extract field”, specifies the requirements for the Extract field, which is fully described in the ZIP Appnote.txt.

Table C–3. Support for Version Needed to Extract field

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Version | Feature | Additional requirements on packages | Additional requirements on consumers |  |
| 1.0 | Default value | No  |  |  |
| 1.1  | File is a volume label | No  | Shall not interpret as a part |  |
| 2.0 | File is a folder (directory) | No  | Shall not interpret as a part |  |
| 2.0 | File is compressed using Deflate compression | No  |  |  |
| 2.0 | File is encrypted using traditional PKWARE encryption | No |  |  |
| 2.1 | File is compressed using Deflate64(tm) | No |  |  |
| 2.5 | File is compressed using PKWARE DCL Implode  | No |  |  |
| 2.7 | File is a patch data set  | No |  |  |
| 4.5 | File uses ZIP64 format extensions |  |  |  |
| 4.6 | File is compressed using BZIP2 compression | No |  |  |
| 5.0 | File is encrypted using DES | No |  |  |
| 5.0 | File is encrypted using 3DES | No |  |  |
| 5.0 | File is encrypted using original RC2 encryption | No |  |  |
| 5.0 | File is encrypted using RC4 encryption | No |  |  |
| 5.1 | File is encrypted using AES encryption | No |  |  |
| 5.1 | File is encrypted using corrected RC2 encryption | No |  |  |
| 5.2 | File is encrypted using corrected RC2-64 encryption | No |  |  |
| 6.1 | File is encrypted using non-OAEP key wrapping | No |  |  |
| 6.2 | Central directory encryption | No |  |  |

Table C–4, “Support for Compression Method field”, specifies the additional requirements for the Compression Method field, which is fully described in the ZIP Appnote.txt.

Table C–4. Support for Compression Method field

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Code | Method | Additional requirements on packages | Additional requirements on consumers |  |
| 0 | The file is stored (no compression) | No  |  |  |
| 1 | The file is Shrunk | No |  |  |
| 2 | The file is Reduced with compression factor 1 | No |  |  |
| 3 | The file is Reduced with compression factor 2 | No |  |  |
| 4 | The file is Reduced with compression factor 3 | No |  |  |
| 5 | The file is Reduced with compression factor 4 | No |  |  |
| 6 | The file is Imploded | No |  |  |
| 7 | Reserved for Tokenizing compression algorithm | No |  |  |
| 8 | The file is Deflated |  |  |  |
| 9 | Enhanced Deflating using Deflate64™ | No |  |  |
| 10 | PKWARE Data Compression Library Imploding | No |  |  |
| 11 | Reserved by PKWARE | No |  |  |

Table C–5, “Support for modes/structures defined by general purpose bit flags”, specifies the additional requirements when utilizing these general-purpose bit flags within records.

Table C–5. Support for modes/structures defined by general purpose bit flags

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Bit | Feature | Requirements on packages | Requirements on consumers |  |
| 0 | The file is encrypted. | Partially — the value shall be set to 0 |  |  |
| 1, 2 |

|  |  |  |
| --- | --- | --- |
| Bit 2 | Bit 1 |  |
| 0 | 0 | Normal (-en) compression option was used. |
| 0 | 1 | Maximum (-exx/-ex) compression option was used. |
| 1 | 0 | Fast (-ef) compression option was used. |
| 1 | 1 | Super Fast (-es) compression option was used. |

 |  |  |  |
| 3 | The fields crc-32, compressed size and uncompressed size are set to zero in the local header. The correct values are put in the data descriptor immediately following the compressed data. |  |  |  |
| 4 | Reserved for use with method 8, for enhanced deflating | Partially — the value shall be set to 0 |  |  |
| 5 | The file is compressed patched data. (Requires PKZIP version 2.70 or greater.) | Partially — the value shall be set to 0 |  |  |
| 6 | You should set the version needed to extract value to at least 50 and you shall set bit 0. If AES encryption is used, the version needed to extract value shall be at least 51. | Partially — the value shall be set to 0 |  |  |
| 7 | Currently unused | Partially — the value shall be set to 0 |  |  |
| 8 | Currently unused | Partially — the value shall be set to 0 |  |  |
| 9 | Currently unused | Partially — the value shall be set to 0 |  |  |
| 10 | Currently unused | Partially — the value shall be set to 0 |  |  |
| 11 | Currently unused | Partially — the value shall be set to 0 |  |  |
| 12 | Unused | Partially — the value shall be set to 0 |  |  |
| 13 | Used when encrypting the Central Directory to indicate selected data values in the Local Header are masked to hide their actual values. See the section describing the Strong Encryption Specification for details. | Partially — the value shall be set to 0 |  |  |
| 14 | Unused | Partially — the value shall be set to 0 |  |  |
| 15 | Unused | Partially — the value shall be set to 0 |  |  |

Table C–6, “Support for Extra field (variable size), PKWARE-reserved”, specifies the additional requirements for the Extra field entries reserved by PKWARE and described in the ZIP Appnote.txt.

Table C–6. Support for Extra field (variable size), PKWARE-reserved

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field ID | Field description | Requirements on packages | Requirements on consumers |  |
| 0x0001 | ZIP64 extended information extra field |  |  |  |
| 0x0007 | AV Info | Partially — the value shall be set to 0 |  |  |
| 0x0008 | Reserved for future Unicode file name data (PFS) | Partially — the value shall be set to 0 |  |  |
| 0x0009 | OS/2 | Partially — the value shall be set to 0 |  |  |
| 0x000a | NTFS  | Partially — the value shall be set to 0 |  |  |
| 0x000c  | OpenVMS | Partially — the value shall be set to 0 |  |  |
| 0x000d | Unix | Partially — the value shall be set to 0 |  |  |
| 0x000e | Reserved for file stream and fork descriptors | Partially — the value shall be set to 0 |  |  |
| 0x000f | Patch Descriptor | Partially — the value shall be set to 0 |  |  |
| 0x0014 | PKCS#7 Store for X.509 Certificates | Partially — the value shall be set to 0 |  |  |
| 0x0015 | X.509 Certificate ID and Signature for individual file | Partially — the value shall be set to 0 |  |  |
| 0x0016 | X.509 Certificate ID for Central Directory | Partially — the value shall be set to 0 |  |  |
| 0x0017 | Strong Encryption Header | Partially — the value shall be set to 0 |  |  |
| 0x0018 | Record Management Controls | Partially — the value shall be set to 0 |  |  |
| 0x0019 | PKCS#7 Encryption Recipient Certificate List | Partially — the value shall be set to 0 |  |  |
| 0x0065 | IBM S/390 (Z390), AS/400 (I400) attributes — uncompressed | Partially — the value shall be set to 0 |  |  |
| 0x0066 | Reserved for IBM S/390 (Z390), AS/400 (I400) attributes — compressed | Partially — the value shall be set to 0 |  |  |
| 0x4690 | POSZIP 4690 (reserved) | Partially — the value shall be set to 0 |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

All 64-bit stream record sizes and offsets shall have the high-order bit = 0.

All fields that contain “number of entries” shall not exceed 2,147,483,647.