05 January 2009/Tommy Schomacker

ISO/DIS 28560 and DS/INF 163

1. Scope

This document summarizes the relations between ISO/DIS 28560 RFID in Libraries and DS/INF 163 RFID Data Model for Libraries.

2. General

DS/INF 163 corresponds to ISO/DIS 28560-1 (General requirements and data elements) and ISO/DIS 28560-3 (Fixed Length encoding) – in the following called the ISO-version. Another encoding is described in ISO/DIS 28560-3 (Encoding based on ISO/IEC 15962).

The aim has been that an RFID-tag encoded according to DS/INF 163 also conforms to ISO/DIS 28560.

But in the editing of ISO/DIS 28560 it has been necessary to make some clarifications and some minor changes.

The main problems can be summarized as follows.

- DSFID. The ISO version mandates a specific value
- Data block ID. The 32 bit-option is not valid in the ISO-version.
- Set information. In the ISO-version this data element covers Number of parts in item and Ordinal part number. If some parts do not have an RFID tag, the ordinal part number of the first part must be zero.

3. AFI

New values have been assigned in the ISO-version. This is covered in DS/INF 163 amendment #2.

4. DSFID

This construct is not mentioned in DS/INF 163.

In ISO/DIS 28560-1 the construct is addressed in section 7.2.2 and 11.4 – see appendix B.

In ISO/DIS 28560-3 the construct is addressed in section 5.1 and 5.3 – see appendix C.

In short the difference is that DS/INF 163 does not specify a value for DSFID while the ISO-version specifies a specific value.

5. Data elements

The data elements are defined in ISO/DIS 28560-1. The following requirements is defined:

"This list of data elements forms an input for the other Parts of this Standard. The data elements in the other Parts of this International Standard shall be compliant with this Part 1. No additional data elements shall be included without an Amendment to this first, abstract Part of the Standard and other Parts need to describe how to store any data element in the RFID tag that is listed above."

ISO/DIS 28560-3 describes the encoding of these data elements. This encoding implicitly defines the mapping to the data elements defined in DS/INF 163.

A number of the data elements in ISO/DIS 28560-1 have no counterpart in DS/INF 163.

Appendix A gives a specific mapping from DS/INF 163 to the ISO-version.

6. Data blocks

ISO/DIS 28560-3 gives the following definition: A data block is a structured sequence of data elements.

The following types of data blocks exists in ISO/DIS 28560-3:

Basic block: data block always occupying the first 256 (or 272) bits of the RFID-tag

Extension block: optional data block following the basic block

Filler data block: optional data block that may be inserted to align other data blocks on page boundaries

End block: data block terminating the information on the RF-tag

ISO/DIS 28560-3 defines four extension blocks:

- Library extension block
- Acquisition extension block
- Library supplement block
- Title block

DS/INF 163	*ISO/DIS 28560*	*Remarks*
Mandatory starting block	Basic block	
End data block	End block	
Filler data block	Filler data block	
Structured extension block	Extension block	
3.5.1 Media format etc. (ID=1)	Library extension block (ID=1)	
3.5.2 Acquisition (ID=2)	Acquisition extension block (ID=2)	This block is augmented in the ISO-version
	Library supplement block (ID=3)	
	Title block (ID=4)	
Unstructured extension block		Has no counterpart in ISO-version

In DS/INF section 3.3.3 is described an optional way to encode the Data block ID using 32 bits. This option is not valid in the ISO-version.

Appendix A. Mapping of data elements

DS/INF 163	*ISO/DIS 28560*	*Remarks*
3.2.1.1 Standard version	2. Content parameter	The value "6" must not be used.
3.2.1.2 Type of usage	5. Type of usage	More values added in ISO-version
3.2.1.3 Number of parts in item 3.2.1.4 Ordinal part number	4. Set information	If some parts do not have an RFID tag, the ordinal part number of the first part must be zero.
3.2.1.5 Primary item ID	1. Primary item identifier	
3.2.1.6 CRC		This field is considered as part of the encoding in the ISO-version.
3.2.1.7 Country of owner library 3.2.1.8 Owner library	3. Owner library (ISIL)	
3.5.1.1 Media format	19. Media format (other)	
3.5.1.2 Alternate item identifier	22. Alternative item identifier	
3.5.1.3 Extended owner library	23. Alternative owner library	
3.5.2.1 Supplier ID	9. Supplier identifier	
3.5.2.2 Item identification	18. Product identifier local	
3.5.2.3 Order number	10. Order number	
3.5.2.4 Invoice number	21. Invoice number	
	6. Shelf location	Not defined in DS/INF 163
	7. ONIX media format	Not defined in DS/INF 163
	8. MARC media format	Not defined in DS/INF 163
	11. ILL borrowing institution (ISIL)	Not defined in DS/INF 163
	12. ILL borrowing transaction number	Not defined in DS/INF 163
	13. Product identifier GS1	Not defined in DS/INF 163
	14. Local data A	Not defined in DS/INF 163
	15. Local data B	Not defined in DS/INF 163
	16. Local data C	Not defined in DS/INF 163
	17. Title	Not defined in DS/INF 163
	20. Supply chain stage	Not defined in DS/INF 163
	24. Subsidiary of an owner library	Not defined in DS/INF 163
	25. Alternative ILL borrowing institution	Not defined in DS/INF 163

Appendix B. ISO/DIS 28560-1 extract

7.2.1 AFI:

The AFI (Application Family Identifier) is a single byte code, often in the systems memory of the tag, used as a tag selection mechanism across the air interface to minimise the extent of communication transaction time with tags that do not carry the relevant AFI code.

The AFI value **$C2_{HEX}$** has been assigned under the registration of ISO/IEC 15961-2 explicitly for library use.

A library may use the AFI in one of two ways:

It may use a single AFI, the value $C2_{HEX}$ assigned under the registration of ISO/IEC 15961-2. This distinguishes library items from all others, and avoids the risk of an RFID reader in another domain reading the RFID tag on a loan item and confusing its encoded content with data in an application outside the library domain. It also enables a library system to reject items that carry a different AFI code, possibly from another domain visited by a client. If a single AFI is used, then a library may wish to lock this. Before they do, they should give consideration to the item's use in other libraries through co-operative arrangements or interlibrary loans. The recipient library may want to use the AFI for security while the item is in their possession even though the donor library does not.
The AFI may additionally be used as part of an "item security system" where the AFI value $C2_{HEX}$ is written to tags for items that are on loan to a client. When the books are returned, an in-stock AFI (07_{HEX}, as agreed by JTC1 SC31 WG4 SG1 to be included in 15961-3) is written to the tag.

RFID tags that comply with this International Standard have a mandatory AFI system data element and should only use one of the two AFI values that are listed above. The AFI together with the Primary Item Identifier and Owner library (ISIL) guarantee global uniqueness of the item.

The AFI has a feature that enables the application to specify that the AFI is to be locked, or leave this unlocked. Once locked, the AFI cannot be unlocked. The AFI shall not be locked if it is used as part of an item security system where two values of the AFI are used, one for on-loan items and one for in-stock items. If other security mechanisms are used, then the AFI may be locked at the individual library's discretion.

Various approaches may be used for securing library items against unauthorised removal. The choice of an item security system is outside the scope of this International Standard and the responsibility of solution providers to develop particular schemes for libraries to choose. Some options are discussed in Annex D, without any comment on their particular merits. Combinations of these options may also be provided in particular systems.

7.2.2 DSFID:

The DSFID (Data Storage Format Identifier) may be present in certain types of tags as a system data element and is the second system data element that is mentioned here. If the DSFID is hard coded - in other words has a specific memory allocation – then the write-DSFID command in the air interface protocol is usually supported by a lock-DSFID command.

If a DSFID is registered and if it is specific for the library domain, the DSFID together with Primary item identifier and Owner library may be used to guarantee global uniqueness. Other Parts of this International Standard should provide further details which system data are used to provide the information that the tag is used for items in the library domain. The DSFID, if present, may also be useful in discriminating between legacy tags and tags that comply with this International Standard.

8.1 Tools for data integrity

When data elements are not programmed in non-volatile memory or are not locked, they can be modified by unauthorised persons. This International Standard does not impose any method for maintaining the integrity of the data, but other Parts of the standard may describe detailed tools for guaranteeing data security. Locking, as discussed for AFI register values in subclause 7.2.1 is one method. This method may also be applied to data stored in tag user memory. Some types of tag hardware may provide additional security tools:

System and / or user data fields can be protected by passwords. Initial writing and / or modifying the data requires the knowledge of a password.
Access to most of the data is blocked when the item is on loan. This is often called the 'privacy mode' of a tag. There is a single command to 'wake up' the tag.

Security tools may be proprietary.

11.3 Using the correct AFI

Even if a decision is taken to postpone migration to this International Standard it is recommended to use the correct AFI code value, certainly for new acquisitions. As many libraries introduced RFID tags before an AFI value for libraries was allocated, the historic installed base has an incorrect AFI value. If the AFI value is not used for security purposes the AFI value is usually zero (00 $_{HEX}$) as test programmes for new tags set this value as the default. Together with other system data elements this zero value for AFI may then be used in migration strategies to discriminate between legacy and ISO compliant tags as is described further on.

A large number of vendor specific RFID tags have a dual AFI value for security purposes. Many different value pairs are in use that are not compliant with the AFI values of subclause 7.2.1. In that case AFI cannot be used for the discrimination between legacy and ISO compliant tags.

11.4 Discrimination between legacy tags and ISO compliant tags

During a changeover or when items from non-compliant libraries mix with those of compliant ones, there is a need to discriminate between a legacy RFID tag and one that is compliant with this International Standard.

Like the AFI value on ‘historic’ tags without a dual AFI value, the DSFID value of a legacy tag is also likely to be zero (00 $_{HEX}$). There is also the possibility that no DSFID field exists on the legacy RFID tags as not all tags have this system data.

The legacy values for AFI or DSFID and the values defined in Part 2, Part 3 and forthcoming Parts should enable, when possible, to discriminate between

— legacy tags that are distributed before AFI was allocated

— legacy tags that are distributed after AFI was allocated (assuming that a correct value is stored on the tag)

— tags that are compliant with a specific part of this International Standard.

The table below indicates the system data for AFI and DSFID on legacy tags compared with tags compliant with this International Standard.

Table 1 — AFI and DSFID values on legacy tags

Type of tag	*AFI value*	*DSFID value*
Legacy tag distributed before AFI was allocated and without a dual AFI-value for item security	Likely to be zero	Likely to be zero or absent
Legacy tag distributed before AFI was allocated and with a dual AFI-value for item security	Any pair of ‘incorrect’ AFI values is possible	Likely to be zero or absent
Legacy tag distributed after AFI was allocated and without a dual AFI-value for item security	$C2_{HEX}$ (possibly locked)	Likely to be zero or absent
Legacy tag distributed after AFI was allocated and with a dual AFI-value for item security	$C2_{HEX}$ (out on loan) and 07_{HEX} (in stock)	Likely to be zero or absent
Tags compliant with this International Standard	As defined in this Part of ISO 28560	As defined in other Parts of ISO 28560

In many cases it will be possible to use one or both of these system data for the discrimination process during a change over period within a library. There are other mechanisms that could be used and implemented in software, but these will be less generic. Advice will be given in other Parts of this International Standard or on the website mentioned in Annex A.

Annex D Interoperability characteristics of security systems

This is an informative annex addressing AFI.

Appendix C. ISO/DIS 28560-3 extract

5.1 Distinguishing from other applications and encodings

The AFI data construct is used to distinguish tags for library applications from other applications.

Tags encoded according to this part of ISO 28560 are distinguished from tags encoded according to ISO 28560-2 by different values of DSFID. If the DSFID data construct does not exist, the tags are distinguished by different values of byte 0.

As described in Part 1, the installed base of legacy tags can be divided in three groups:

- Tags that are compatible with ISO/IEC 18000-3 mode 1 and with AFI not used for item security

- Tags that are compatible with ISO/IEC 18000-3 mode 1 and with AFI used for item security

- Other tags

Tags belonging to the first two groups will usually have AFI = 0 or will use two AFI values for item security. In both cases the AFI value will differ from the value in tags encoded according to this part of ISO 28560.

Readers may use this property to distinguish from these tags.

If legacy tags are not distinguished by the value of AFI, they will eventually be recognized by the CRC-check.

5.2 Writing/reading direction

Writing (and reading) direction is from left to right on the tag – with lowest memory location at the left.

5.3 System area

The value of AFI is defined in ISO 28560-1.

If DSFID is present in system memory as a programmable register, it shall be programmed with the value XX.

6.5 Structured extension blocks

6.5.1 General

Structured extension blocks are used for optional information.

They use a frame structure of 4 bytes to specify length, kind and checksum. The general structure of the extension blocks:

<Length ><Data block ID><XOR checksum>(<data element><end block>)*

It is legal to have two or more extension blocks of the same kind (Data block ID).

Data elements have to occur in the order specified for the specific extension block. A data element may be omitted. In this case the corresponding field must be filled with chr(0).

It is legal to limit the block size by any length. This means that data fields may be truncated or omitted. If so the missing data is assumed to be chr(0).

It is also legal to specify a larger block size than the data requires. If so the redundant part of the block is filled with chr(0).

6.5.1.1 Length

Description:

This field includes the length of this data block. This length includes all bytes in the extension block, including the length itself.
Using 0 or 1 as length has a special meaning (0: End data block, see 6.3.1; 1: Filler data block, see 6.3.2).
Encoding:

8 bit unsigned integer

6.5.1.2 Data block ID

Description:

This field identifies the structured extension block
Values:

1: Library extension block
2: Acquisition extension block
3: Library supplement block
4: Title block
5 to 100: Other extension blocks (for future use)
Encoding:

16 bit unsigned integer (lsb stored at the lowest memory location).

6.5.1.3 Checksum

Description:

For calculating the checksum: Set the checksum field to chr(0), then calculate the XOR for all bytes including length, id and checksum field. Store this value in the checksum field.
For verifying the checksum: calculate the XOR for all bytes including length, id and checksum field, it must be chr(0).