NATIONAL DEPARTMENT OF TRANSPORT (NDOT)
AUTOMATED FARE COLLECTION (AFC) DATA STRUCTURE

NDOT AFC DATA STRUCTURE SPECIFICATION
Version 1.0 Revision 0.5

DOCUMENT CONTROL SHEET

Document Title:	**NDOT AFC Data Structure V1.0 R0.5.docx**	
Date:	25 October 2012	
Short Description:	This document describes the NDOT AFC Data Structure Functionality.	
Contact:	Certification@techso.co.za	
Prepared by:	**Techso (Pty) Ltd** Box 35, Mark Shuttleworth Street The Innovation Hub 0087 South Africa	
In association with:	**Keith Smith Consulting CC** PO Box 66055 Highveld 0169 south Africa	
	Name	**Contact**
Compiled by:	Mr Kobus Lombard	Kobus@techso.co.za
Compiled by:	Mr Keith Smith	Keith.Smith@telkomsa.net
Reviewed by:	Dr Johann Andersen	Johann@techso.co.za

1 INTRODUCTION

Public transport plays a significant role in the social and economic development of South Africa. It also has a direct influence on determining the quality of life of the majority of South African citizens. The number of commuters reliant on public transportation in South Africa is expanding, and significant investment has been earmarked by Government to improve and modernize public transport services.

The South African National Department of Transport (NDOT) has embarked on strategic objectives to transform South African public transport services to an integrated cohesive Automatic Fare Collection (AFC) system. The promulgation of the AFC requirements are designed to establish integrated customer centric public transport services and to simplify the commuter's experience by utilising bank issued payment media carrying additional data fields as specified by NDOT and for use by the public transport industry for automated fare collection.

The NDOT has appointed Techso (Pty) Limited ("the Employer") to establish a Compliance Agency and implement the necessary procedures to certify relevant public transport card reader infrastructure and bank issued cards for AFC Data Structure functionality.

The AFC Datastructure contains the following frames:

- 1 x Secure Card Frame (SCF)
- 3 x Transit Product Frames. Each Transit Product consists of a secure fixed frame (SFF) and variable value frame (VFF)
- 1 x Generic Variable Frame (GVF)

1.1 Related Publications

The following publications contain information related to the contents of this document:

MasterCard

- PayPass M/Chip Issuer Implementation Guide
- PayPass M/Chip Personalization Data Specification
- PayPass M/Chip Technical Specification
- MasterCard Pre-authorised solutions guide 2005
- Version 9 Transit Specification

The full specification suite is subject to the application for and approval of a PayPass licence and is available for the Mchip Advance Platform as well

Limited documentation is available on www.paypass.com

Visa

Visa Transit Payment Specification – Terminal Extension version 1.2 November 2010

Visa Transit Payment Specification – Card Extension version 1.2 November 2010

Visa Transit Payment Specification 1.2 version 1 Updates List Version 1 September 2011

2 CHANGE RECORD

Version 1 Revision 0.4

- Changed references to Visa VIUDS to Visa VTPS

Version 1 Revision 0.5

- Changed references to “VISA” with “Visa”
- Update Visa Transaction flow with correct EMV reference

3 NDOT AFC DATA STRUCTURE

Figure 1 depicts the logical structure to the data elements. The Secure Card Frame will hold card specific information, such as which version of the data structure is currently in use on the card, which other frames are also in use or available, by whom they are used, and where they are located. This Secure Card Frame will be automatically retrieved at start-up by a public transport smartcard proximity coupling device. After the Secure Card Frame has been retrieved the remaining Frames can be retrieved through location and platform independent identifiers within the Secure Card Frame. Seven (7) platform independent location identifiers are included within the Secure Card Frame, six (6) of those are made up of three sets of Value Frames. A Value Frame set consists of a single Fixed Frame and a single Variable Frame. A seventh Frame is for the Generic Variable Frame which will keep most of the Tap On / Tap Off information if required.

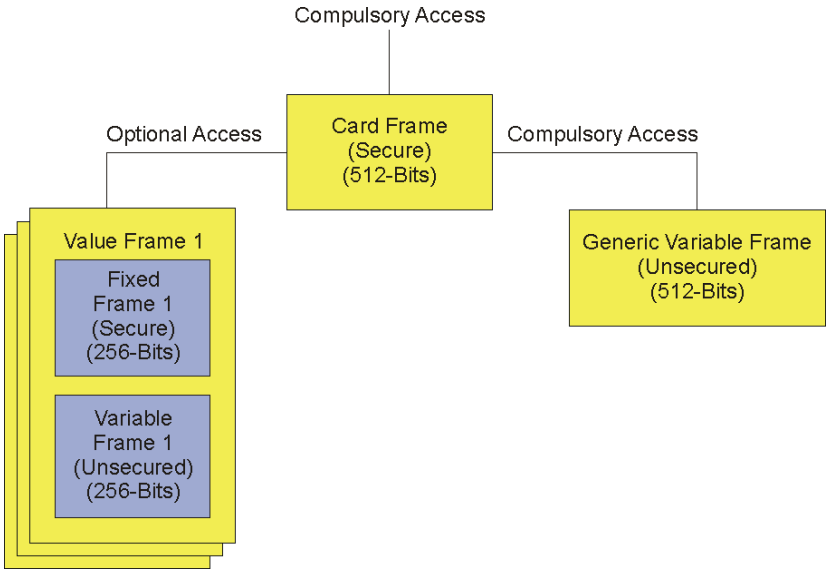


Figure 1: Logical Data Structure Layout

A Value Frame contains information regarding a specific Concession or Ticket that a commuter has procured on his/her card. Each Commuter can procure three (3) different Concessions or Tickets in the current structure. A Fixed Frame holds Concession - / Ticket Specific Information that needs to be secured and the Variable Frame Contains Tap On / Tap Off related information. Table 1 outlines each frame's requirements and the associated data elements, their size, and what they are used for.

It should be noted that the three (3) Value Frames and their contents are not compulsory for a transaction and only contain information regarding a product that can be redeemed if the public transport services allows for such redemption (Monthly Tickets etc.). The Value Frame consists of two parts, a secured frame (Fixed) and an unsecured frame (Variable).

3.1 Card Frame

No	Parameter	Size (Bits)	Type	Description
1.	Frame Revision	8	BIN	Public Transport Data Structure Revision – Can be updated through a mutual PASA / NDOT agreement. (Currently Revision 0x01).
2.	Value Frame Map	3	BITMAP	Each bit indicates which Fixed / Variable Frame are currently in use.
3.	Commuter Class Map	13	BITMAP	Each bit will represent a type of commuter (disabled, elderly, scholar etc.) enabling possible future multiple commuter discounts. A commuter is required to provide Citizenship and ID fields when registering for concessions.
4.	Commuter Citizenship	16	ASCII	ISO 3166 – 1 (alpha-2 ASCII) (If blank commuter is anonymous and regarded as unbanked or unregistered)
5.	Commuter ID	48	BIN	Binary value corresponding to the National Identification/Insurance/Passport Number. (If blank commuter is anonymous and regarded as unbanked or unregistered).
6.	Value Frame 1 Operator ID	96	See 3.10	Operator that has ownership of Value Frame 1
7.	Value Frame 2 Operator ID	96	See 3.10	Operator that has ownership of Value Frame 2
8.	Value Frame 3 Operator ID	96	See 3.10	Operator that has ownership of Value Frame 3.
9.	Secure Fixed Frame 1 Location	16	BIN	TAG / SFI Position. If SFI Usage is false, data holds Tag value. If Tag value is one byte, leading byte is zero. If SFI Usage is true, first byte holds SFI in most significant 5 bits with lower 3 bits set to '100'. Second byte contains record number.
10.	Unsecure Variable Frame 1 Location	16	BIN	TAG / SFI Position. If SFI Usage is false, data holds Tag value. If Tag value is one byte, leading byte is zero. If SFI Usage is true, first byte holds SFI in most significant 5 bits with lower 3 bits set to '100'. Second byte contains record number.
11.	Secure Fixed Frame 2 Location	16	BIN	TAG / SFI Position. If SFI Usage is false, data holds Tag value. If Tag value is one byte, leading byte is zero. If SFI Usage is true, first byte holds SFI in most significant 5 bits with lower 3 bits set to '100'. Second byte contains record number.
12.	Unsecure Variable Frame 2 Location	16	BIN	TAG / SFI Position. If SFI Usage is false, data holds Tag value. If Tag value is one byte, leading byte is zero. If SFI Usage is true, first byte holds SFI in most significant 5 bits with lower 3 bits set to '100'. Second byte contains record number.
13.	Secure Fixed Frame 3 Location	16	BIN	TAG / SFI Position. If SFI Usage is false, data holds Tag value. If Tag value is one byte, leading byte is zero. If SFI Usage is true, first byte holds SFI in most significant 5 bits with lower 3 bits set to '100'. Second byte contains record number.

No	Parameter	Size (Bits)	Type	Description
14.	Unsecure Variable Frame 3 Location	16	BIN	TAG / SFI Position. If SFI Usage is false, data holds Tag value. If Tag value is one byte, leading byte is zero. If SFI Usage is true, first byte holds SFI in most significant 5 bits with lower 3 bits set to '100'. Second byte contains record number.
15.	Unsecure Generic Variable Frame Location	16	BIN	TAG / SFI Position. If SFI Usage is false, data holds Tag value. If Tag value is one byte, leading byte is zero. If SFI Usage is true, first byte holds SFI in most significant 5 bits with lower 3 bits set to '100'. Second byte contains record number.
16.	SFI Usage Map	8	BITMAP	Each bit indicates which Frame Location is currently True for SFI/Record Usage, else false for TAG usage (First bit (least significant bit) corresponds to item 9 within this list and the seventh bit corresponds to item 15). The last bit (most significant bit) must always be set to 0
17.	Reserved for Future Use	16	N/A	Reserved for Future Use

3.2 Product Fixed Frame

No	Parameter	Size (Bits)	Type	Description
18.	Product / Special Event ID	24	BIN	Type of Concession / Ticket this Frame will hold.
19.	Product / Special Event Unique Instance ID	32	BIN	Unique instance ID of the product in question.
20.	Start Date for Frame	24	BCD	09/01/28 (YY/MM/DD, without the “/”)
21.	End Date for Frame	24	BCD	09/01/28 (YY/MM/DD, without the “/”)
22.	Control Bits	16	BITMAP	Product specific bits (IFM product specific)
23.	Reserved for Future Use	136	N/A	Reserved for Future Use

3.3 Product Variable Frame

No	Parameter	Size (Bits)	Type	Description
24.	Control Bits	16	BITMAP	IFM product specific
25.	Transaction Sequence ID	16	BIN	Transaction sequence possibility if multi-operator multi-modal is available for current frame.
26.	Last Date Usage	24	BCD	09/01/28 (YY/MM/DD, without the “/”) When last was the product used.

No	Parameter	Size (Bits)	Type	Description
27.	Daily In Counter / Counter	12	BIN	Dual usage counter (6 bits counter, other 6 bits limit) or 12 bit Binary
28.	Daily Out Counter / Counter	12	BIN	Dual usage counter (6 bits counter, other 6 bits limit) or 12 bit Binary
29.	MAC	64	BIN	Message Authentication Code (MAC). Retail MAC / ISO 9797-1 Information Technology - Message Authentication Codes - MAC Algorithm 3 - Padding Method 2
30.	Reserved for Future Use	112	N/A	Reserved for Future Use

3.4 Generic Variable Frame

No	Parameter	Size (Bits)	Type	Description
31.	Boarding Date & Time	40	BCD	09/01/28.16:05 (YY/MM/DD.HH:MM, without the "/","." , and ":")
32.	Boarding Location Type	2	BIN	0 = Informal, 1 = Bus/Taxi Stops, 2 = Facility, 3 = Route
33.	Boarding Route ID / Facility ID	96	ASCII	See Table , Table, and Table for the encodings associated with Boarding Location Type.
34.	Boarding GPS Longitude / Zone / KM	32	FLOAT	IEEE 754-1985 - Single Precision
35.	Boarding GPS Latitude / Zone / KM	32	FLOAT	IEEE 754-1985 - Single Precision
36.	Boarding Value Frame	2	BIN	Which Variable / Fixed Frame were used if any?
37.	Boarding Control Bits	16	BITMAP	IFM product specific
38.	Alighting Date & Time	40	BCD	09/01/28.16:05 (YY/MM/DD.HH:MM, without the "/","." , and ":")
39.	Alighting Location Type	2	BIN	0 = Informal, 1 = Bus/Taxi Stops, 2 = Facility, 3 = Route
40.	Alighting Route ID / Facility ID	96	ASCII	See Table , Table, and Table for the encodings associated with Alighting Location Type.
41.	Alighting GPS Longitude / Zone / KM	32	FLOAT	IEEE 754-1985 - Single Precision
42.	Alighting GPS Latitude / Zone / KM	32	FLOAT	IEEE 754-1985 - Single Precision
43.	Alighting Value Frame	2	BIN	Which Variable / Fixed Frame were used if any?
44.	Alighting Control Bits	16	BITMAP	IFM product specific

No	Parameter	Size (Bits)	Type	Description
45.	MAC	64	BIN	Message Authentication Code (MAC). Retail MAC / ISO 9797-1 Information Technology - Message Authentication Codes - MAC Algorithm 3 - Padding Method 2
46.	Reserved for Future Use	8	N/A	Reserved for Future Use

3.5 Coding of Public Transport Stops

Data Element	Byte Position	Value Format	Description
Country	XXOOOOOOOOOO	ISO 3166 – 1 (alpha-2 ASCII)	South Africa = 'ZA'
Province	OOXXOOOOOOOO	ISO 3166 – 2 (subdivision code 2 characters in ASCII)	Gauteng = 'GP'
Municipality Type	OOOOXOOOOOO	ASCII	(L)ocal, (D)istrict, and (M)etropolitan
Municipality	OOOOOXXOOOO	BIN	0 - 65535
Stop Type	OOOOOOOXOOO	ASCII	(F)ormal, and (I)nformal
Stop	OOOOOOOOXXO	BIN	0 - 65535

Table 1: Coding of Public Transport Stops

3.6 Coding of Public Transport Facilities

Data Element	Byte Position	Value Format	Description
Country	XXOOOOOOOOOO	ISO 3166 – 1 (alpha-2 ASCII)	South Africa = 'ZA'
Province	OOXXOOOOOOOO	ISO 3166 – 2 (subdivision code 2 characters in ASCII)	Gauteng = 'GP'
Municipality Type	OOOOXOOOOOO	ASCII	(L)ocal, (D)istrict, and (M)etropolitan
Municipality	OOOOOXXOOOO	BIN	0 - 65535
Facility Type	OOOOOOOXXOO	ASCII	TR = Minibus-Taxi Rank TH = Minibus –Taxi Holding BT = Bus Terminal BD = Bus Depot BH = Bus Holding RS = Railway Station PR = Park & Ride MR = Metered Taxi Rank AP = Airport SP = Seaport
Facility	OOOOOOOOOXOO	BIN	0 - 255

Table 2: Coding of Public Transport Facilities

3.7 Coding of Public Transport Routes

Data Element	Byte Position	Value Format	Description
Country	XXOOOOOOOOOO	ISO 3166 – 1 (alpha-2 ASCII)	South Africa = 'ZA'
Province	OOXXOOOOOOOO	ISO 3166 – 2 (subdivision code 2 characters in ASCII)	Gauteng = 'GP'
Municipality Type	OOOOXOOOOOOO	ASCII	(L)ocal, (D)istrict, and (M)etropolitan
Municipality	OOOOOXXOOOOO	BIN	0 - 65535
Route Direction	OOOOOOOXOOOO	ASCII	(I)ngoing, (O)utgoing, (F)orward, (R)eturn, (N)orth, (E)ast, (S)outh, or (W)est
Operator	OOOOOOOOXOOO	BIN	
Transport Type	OOOOOOOOOXOO	ASCII	(B)us, Minibus-(T)axi, (R)ail, (A)ir, and (S)ea
Route	OOOOOOOOOOXX	BIN	0 - 65535

Table 3: Coding of Public Transport Routes

3.8 Coding of Operator ID

Data Element	Byte Position	Value Format	Description
Country	XXOOOOOOOOOO	ISO 3166 – 1 (alpha-2 ASCII)	South Africa = 'ZA'
Province	OOXXOOOOOOOO	ISO 3166 – 2 (subdivision code 2 characters in ASCII)	Gauteng = 'GP'
Municipality Type	OOOOXOOOOOOO	ASCII	(L)ocal, (D)istrict, and (M)etropolitan
Municipality	OOOOOXXOOOOO	BIN	0 - 65535
Operator	OOOOOOOXXXXX	ASCII	Unique abbreviations

3.9 General

- MAC value is calculated using ISO9797-1 MAC Algorithm 3 Padding Method 2
- The data for MAC calculation must consist of
  - the card's unique number
  - the card's expiry date
  - frame transaction counter *(Note: currently supported by Visa only)*
  - the contents of the specific NDOT AFC datastructure frame
- Each AFC vendor must implement their own certificate/key management processes/technology
- Should product interoperability (i.e. multiple operators sharing same product) be required, the relevant AFC vendors will have to share information/keys to enable sharing of products and calculation/verification of MAC's for the shared product frames and boarding/alighting frame in the NDOT AFC datastructure. This sharing of information/keys between the different AFC vendors must be facilitated by the relevant operators
- The Compliance Agency will not check for the correct calculation of the MAC value
- Algorithm:

- Take the data in the relevant frame and exclude the MAC
- Pad the data using the MAC algorithm 3 Padding method 2
- Retrieve the MAC UDK if it exists, else derive it from the MAC MDK
- Set the Ciphermode to ECB and the Padding mode to none (it has already been padded)
- Start with an initial value = “00000000000000” (hex)
- Cut the padded data in “XXXXXXXXXXXXXXXX” (hex) blocks
- Use the XOR operator on the initial value and the first data block, encrypt this block with the left part of the UDK using the DES algorithm
- The initial value becomes now the result of the calculation above.
- Repeat the previous two steps over and over again until all data blocks have been used.
- The final result is the MAC

4 MASTERCARD MCA/ADVANCED SMART CARD

The following information is for reference purposes only. MasterCard Worldwide will provide detailed technical information at the signing of relevant Non-Disclosure Agreement.

4.1 COMMANDS

The transit application data will be stored in data objects and/or files. Access to the transit data will be through:

1. GET DATA or PUT DATA commands which respectively read data from data objects and write data to data objects; or
2. READ RECORD or UPDATE RECORD commands using a SFI (Short File Identifier) & Record reference. Typically the content of the record itself will consist of data object(s).

A reference to the above four commands can be found in the following specifications:

1. SABS ISO/IEC 7816-4:1995 or later edition(s), and
2. The M/Chip 4 Version 1.1 Card Application Specifications for Debit and Credit, dated October 2006.

4.2 DATA OBJECT – BER-TLV STRUCTURE

As defined in ISO/IEC 8825 a BER-TLV (Basic Encoding Rules – Tag Length Value) data object consists of three parts:

- A tag uniquely identifies a data object within the environment of an application;
- The length is the length of the value field of the data object; and
- The value of the data object will consist of a data element.

4.3 STORAGE OF DATA STRUCTURE

4.3.1 Storage of Secure Card Frame

The Secure Card Frame will always be referenced by Tag 9F7B (for local South African conditions, internationally accepted Tag would require MasterCard to provide an available Tag for use). Tag 9F7B (or international tag) will be exactly 64 bytes in length. Further Tag references within the value part of Tag 9F7B (or international Tag) are Issuer dependent. The fields in the Secure Card Frame as shown in the following two tables will contain a 2 byte Tag reference. It is very important that the SFI usage map (Element number 16 in **Error! Reference source not found.**) be 0b00000000 for he MasterCard MICA platform. The SFI usage map's usage will be updated by MasterCard Worldwide for the M/Chip Advance Specification.

For example, a Secure Card Frame (which is discussed later) containing Tag references:

Field Name	Bit Size	Type	Value
Frame Revision	8	Binary	01_{16}
Value Frame Map	3	BITMAP	
Commuter Class Map	13	BITMAP	
Commuter Citizenship	16	ASCII	
Commuter ID	48	Binary	
Value Frame 1 Operator ID	96	ASCII	
Value Frame 2 Operator ID	96	ASCII	
Value Frame 3 Operator ID	96	ASCII	
Secure Fixed Value Frame 1 location	16	Binary	$9F70_{16}$
Unsecure Variable Value Frame 1 location	16	Binary	9F75
Secure Fixed Value Frame 2 location	16	Binary	$9F71_{16}$
Unsecure Variable Value Frame 2 location	16	Binary	9F76
Secure Fixed Value Frame 3 location	16	Binary	$9F72_{16}$
Unsecure Variable Value Frame 3 location	16	Binary	$9F77_{16}$
Unsecure Generic Variable Frame	16	Binary	$9F78_{16}$
SFI usage map	8	Boolean	00000000_2
RFU (Reserved for Future Use)	16	N/A	

Tabel 4: MasterCard Secure Card Frame containing Tag references – Example

4.3.2 Storage of Secure Fixed Value Frame

- Provision is made for three (3) Secure Fixed Frames.

- Each Secure Fixed Frame will be exactly 32 bytes in length.
- Each Secure Fixed Frame will be stored under its own Tag.
- Each Issuer need to select their own Tag references. These references need to be stored in the Secure Card Frame.
- The Secure Fixed Frame will require Secure Messaging to allow update in an online contact environment. No additional security measures are required from the Transport industry for this Frame.

4.3.3 Storage of Unsecure Generic Variable Frame

- Provision is made for one (1) Unsecure Generic Variable Frame.
- The Unsecure Generic Variable Frame will be exactly 64 bytes in length.
- The Unsecure Generic Variable Frame Each will be stored under a Tag.
- Each Issuer need to select their own Tag reference. This reference needs to be stored in the Secure Card Frame.
- The Unsecure Generic Variable Frame will not require Secure Messaging, and thus will be allowed to be modified in an offline contactless environment. The provision of a Transport MAC will be allowed for this Frame, but are not supplied by the Card Association.

4.3.4 Storage of Unsecure Variable Value Frame

- Provision is made for three (3) Unsecure Variable Frames.
- Each Unsecure Variable Frame is exactly 32 bytes in length.
- Each Unsecure Variable Frame will be stored under its own Tag.
- Each Issuer need to select their own Tag reference. This reference needs to be stored in the Secure Card Frame.
- The Unsecure Variable Frames will not require Secure Messaging, and thus will be allowed to be modified in an offline contactless environment. The provision of a Transport MAC will be allowed for each Frame, but are not supplied by the Card Association.

4.4 READING OF TRANSIT DATA

4.4.1 Secure Card Frame

Free read access (non-secure) through the use of a READ RECORD command. It is suggested that the Tag 9F7B (for local South African conditions, internationally accepted Tag would require MasterCard to provide an available Tag for use) be stored in one of the SFIs 1 through 10, and ensure that the SFI/Record combination is included in the contactless application's application file locator. It is also highly recommended that the SFI/Record combination is placed to be read first within the AFL records during Card Personalisation. This will enable the Secure Card Frame to be read along with the remainder of the EMV Tags required for terminal risk management and action analysis.

4.4.2 Secure Fixed Frames

- Free read access (non-secure) through the use of a GET DATA command.
- Actual Tag references will be stored in the Secure Card Frame.

4.4.3 Unsecure Generic Variable Frame

- Free read access (non-secure) through the use of a GET DATA command.
- Actual Tag references will be stored in the Secure Card Frame.

4.4.4 Unsecure Variable Frame

- Free read access (non-secure) through the use of a GET DATA command.

- Actual Tag references will be stored in the Secure Card Frame.

4.5 WRITING OF TRANSIT DATA

Access Protected Write (Secure):

Transit data can be written to Secure Card Frame, and Secure Fixed Frames, only once a special access condition had been fulfilled. For write access to Secure Fixed Frames, it is necessary to cryptographically secure the data transmission to the chip card to prevent unauthorised manipulation of the transit data. This sort of security for chip cards is called Secure Messaging. It involves adding a MAC (message authentication code) to each PUT DATA or UPDATE RECORD command. This requires knowledge of a session key between the card and the Issuing Bank's host system. Only the Issuing Bank has knowledge of the Master Keys. This process also requires an online transaction and will be a contact-based chip transaction.

Secure Card Frame

- Access protected write (secure) to Tag 9F7B (or international equivalent) through the use of an UPDATE RECORD command.

Secure Fixed Frames

- Access protected write (secure) through the use a PUT DATA command.
- Actual Tag references will be stored in the Secure Card Frame.

Unsecure Generic Variable Frame

- Free write access (non-secure) through the use of a PUT DATA command.
- Actual Tag references will be stored in the Secure Card Frame.

Unsecure Variable Frame

- Free write access (non-secure) through the use of a PUT DATA command.
- Actual Tag references will be stored in the Secure Card Frame.

4.6 TRANSACTION FLOWS

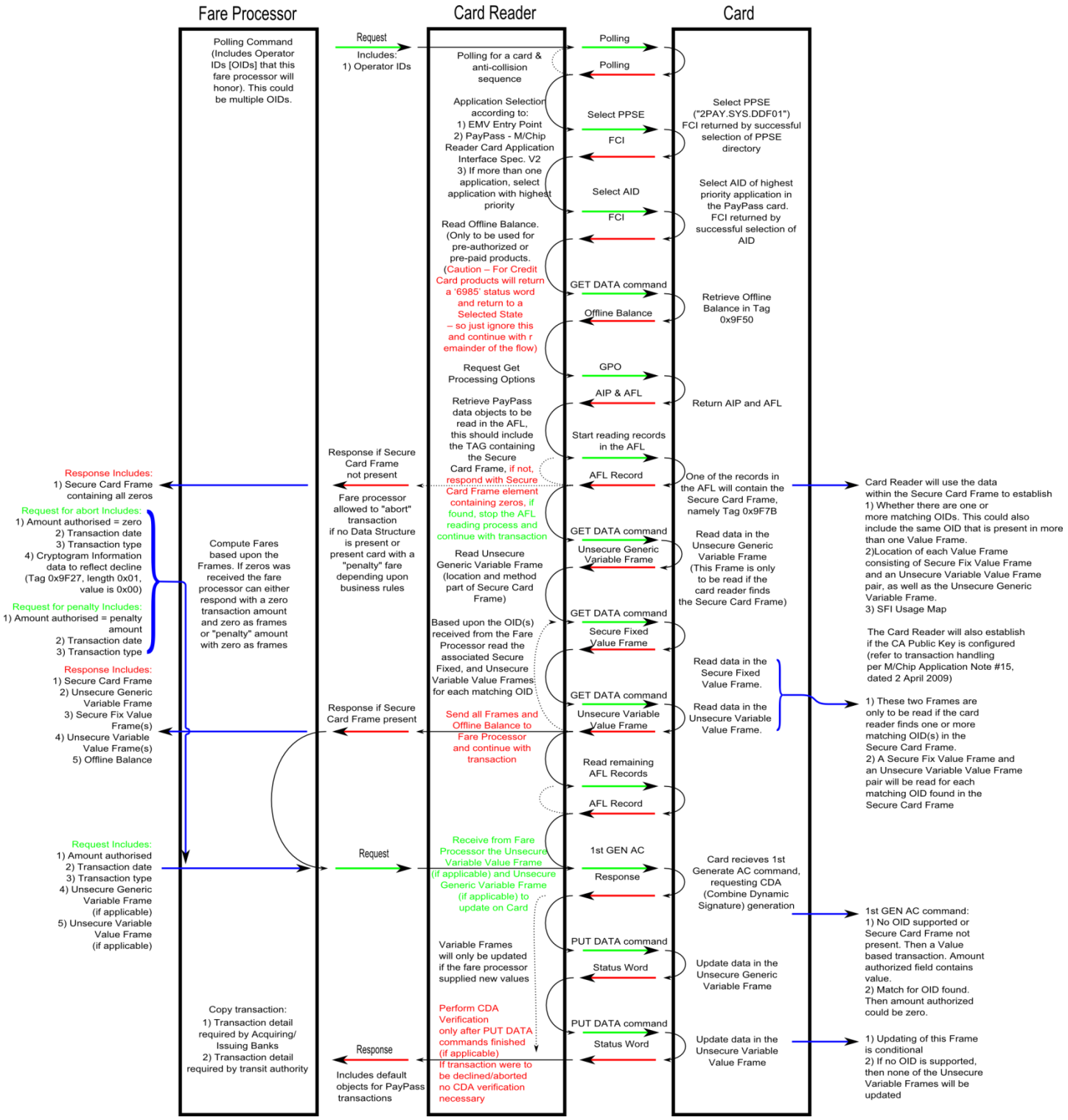


Figure 2: Process flow of a MICA/Advanced Smart Card

Please contact MasterCard South Africa for more information.

5 VISA VTPS SMART CARD

Please contact Visa South Africa for more information.

Visa Inc's detailed technical specifications can be provided on signing of relevant agreements between interested parties and Visa.

The reference documents containing detailed information relevant for the Data Structure are called:

- Visa Transit Payment Specification – Terminal Extension version 1.2 November 2010
- Visa Transit Payment Specification – Card Extension version 1.2 November 2010
- Visa Transit Payment Specification 1.2 version 1 Updates List Version 1 September 2011

These specifications work in combination with Visa standard EMV payment specifications and are referred to in this Appendix as the Visa Transit Payment Specifications.

5.1 COMMANDS

The transit application data will be stored in fixed linear record files and can be read and updated via the commands described in the Visa Transit Payment Specifications.

5.2 STORAGE OF DATA STRUCTURE

5.2.1 Storage of Secure Card Frame

The Secure Card Frame and Unsecure Generic Variable Frame will always be referenced by Scheme ID 17 10 00 01_{16} .(used as Scheme ID for the compulsory Frames for the South African ticket structure).

The Scheme ID 17 10 00 01_{16} will reference the 64 bytes of data for the Secure Card Frame (as defined in chapter 2), which are updateable only via secure messaging as defined in EMV plus the 64 bytes of data for the Unsecure Generic Variable Frame (as defined in chapter 2), which are updateable by a transit operator terminal. Further Scheme ID references within the contents of the Issuer Secured Data part of Scheme ID 17 10 00 01_{16} are as defined in section 0 and as agreed upon between Issuer and transit operator. The fields in the Secure Card Frame as shown in the following table will contain a 2 byte Scheme ID, which can be used to reference an actual file and record location. The contents of the 2 bytes location shall for South African transit ticket structure always be preceded by the digits 17 10 in order to form the final Scheme ID. For further details in how to reference Scheme ID's with actual record and file location see the Visa Transit Payment Specifications.

For example, a Secure Card Frame containing Scheme ID references:

Field Name	Bit Size	Type	Value
Frame Revision	8	Binary	01_{16}
Value Frame Map	3	BITMAP	
Commuter Class Map	13	BITMAP	
Commuter Citizenship	16	ASCII	
Commuter ID	48	Binary	
Value Frame 1 Operator ID	96	ASCII	
Value Frame 2 Operator ID	96	ASCII	
Value Frame 3 Operator ID	96	ASCII	
Secure Fixed Value Frame 1 location	16	Binary	0002_{16}
Unsecure Variable Value Frame 1 location	16	Binary	0002_{16}
Secure Fixed Value Frame 2 location	16	Binary	0003_{16}
Unsecure Variable Value Frame 2 location	16	Binary	0003_{16}
Secure Fixed Value Frame 3 location	16	Binary	0004_{16}
Unsecure Variable Value Frame 3 location	16	Binary	0004_{16}
Unsecure Generic Variable Frame	16	Binary	0001_{16}
SFI usage map	8	Boolean	01111111_{2}
RFU (Reserved for Future Use)	16	N/A	

Table 5: Visa Secure Card Frame containing Scheme ID references – Example

5.2.2 Storage of Secure Fixed Value Frame and Unsecure Variable Value Frame

Provision is made for three (3) Secure Fixed Value Frame/Unsecure Variable Value Frame pairs.

Each Secure Fixed Value Frame/Unsecure Variable Value Frame pair will be referenced by Scheme on of the scheme ID's 17 10 00 02_{16}, 17 10 00 03_{16}, or 17 10 00 04_{16} (used as Scheme ID for the optional Frames for the South African ticket structure).

Each optional Scheme ID will reference the 32 bytes of data for the Secure Fixed Value Frame (as defined in chapter 2), which are updateable only via secure messaging as defined in EMV plus the 32 bytes of data for the Unsecure Variable Value Frame (as defined in chapter 2), which are updateable by a transit operator terminal For further details in how to reference Scheme ID's with actual record and file location see the Visa Transit Payment Specifications.

5.3 READING OF TRANSIT DATA

5.3.1 Secure Card Frame

- Free read access (non-secure) through the use of the READ RECORD command as defined in the Visa Transit Payment Specification.
- The Secure Card Frame will always be referenced by Scheme ID 17 10 00 01_{16} by all South African card issuers.
- The record read will contain both Secure Card Frame and Unsecure Generic Variable Frame.

5.3.2 Secure Fixed Value Frames

- Free read access (non-secure) through the use of the READ RECORD command as defined in the Visa Transit Payment Specification.
- Scheme ID 17 10 00 02_{16}, 17 10 00 03_{16}, or 17 10 00 04_{16} are available for Secure Fixed Value Frame's.
- The record read will contain both Secure Fixed Value Frame and Unsecure Variable Value Frame

5.3.3 Unsecure Variable Value Frame

- Free read access (non-secure) through the use of a READ RECORD command as defined in the Visa Transit Payment Specification.
- Scheme ID 17 10 00 02_{16}, 17 10 00 03_{16}, or 17 10 00 04_{16} are available for Unsecure Variable Value Frame's
- The record read will contain both Secure Fixed Value Frame and Unsecure Variable Value Frame

5.3.4 Unsecure Generic Variable Frame

- Free read access (non-secure) through the use of a READ RECORD command as defined in the Visa Transit Payment Specification.
- Scheme ID 17 10 00 01_{16}, is used to reference the Unsecure Generic Variable Frame.
- The record read will contain both Secure Card Frame and Unsecure Generic Variable Frame

5.4 WRITING OF TRANSIT DATA

5.4.1 Write of Secure Frames

Transit data can be written to Secure Card Frame, and Secure Fixed Value Frames, only once a special access condition had been fulfilled. For write access to Secure Card Frame and Secure Fixed Value Frames, it is necessary to cryptographically secure the data transmission to the chip card to prevent unauthorised manipulation of the transit data. This sort of security for chip cards is defined in EMV and is called Secure Messaging. It involves adding a MAC (Message Authentication Code) to each command updating the data. This requires knowledge of a session key between the card and the Issuing Bank's or transit ticket processor host system. Only the Issuing Bank or a transit ticket processor designated by the issuing bank has knowledge of the Master Key(s) used to derive the session key. This process therefore requires an online transaction. The transaction can be conducted over the contact or contactless interface.

Secure Card Frame

- Access protected write (secure) to Scheme ID 17 10 00 01_{16} through the use of an update command as defined in the Visa Transit Payment Specification.

Secure Fixed Value Frames

- Access protected write (secure) through the use of an update command as defined in the Visa Transit Payment Specification.
- Notice that in addition an update of actual Scheme ID or other information (e.g. Operator ID) in Secure Card Frame might be required.

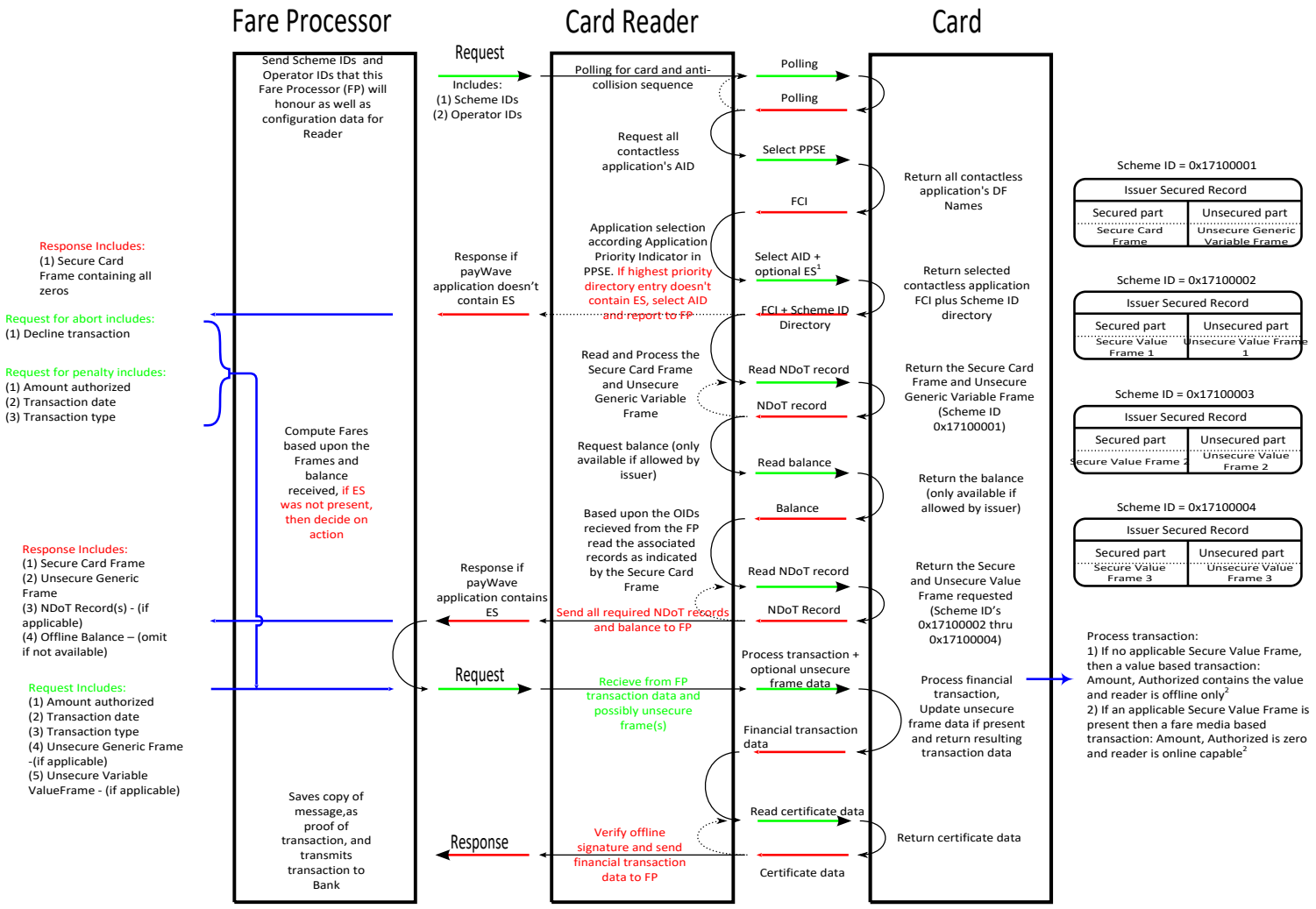
5.4.2 Write of Unsecure Frames

Visa Transit Payment Specifications caters for two methods of updating Unsecure Generic Variable Frame and Unsecure Variable Value Frame:

- Free write access (non-secure) through the use of an update commands as define in the Visa Transit Payment Specifications.
- Limited write lock access through the use of an update command and mechanism as described in the Visa Transit Payment Specifications. The lock mechanism) can use a Transit operator implementation specific key (Not to be supplied by the issuer).

5.5 TRANSACTION FLOWS

Figure 3 Process flow of a VISA Smart Card



[1] ES stands for Extended Selection (tag 9F29), as defined in EMV Contactless Specifications, Book B, 2.2, June 2012

[2] Exact coding as outlined in Visa technical specification