EMV Specification
EMV Specifications

- May ‘94 - Version 1.0 EMV Part 1
- Aug ‘94 - Version 1.0 EMV Part 2
- Oct ‘94 - Version 1.0 EMV Part 3
- Jun ‘95 - Version 2.0 EMV
- Jun ‘96 - Version 3.0 EMV’96
- May ‘98 - Version 3.1.1
- Dec ‘2000 - EMV2000 (Version 4.0)
EMV Versions 1 & 2

- Divided into 3 parts:
  - Part 1: Electromechanical Characteristics, Logical Interface & Transmission Protocol
  - Part 2: Data Elements & Commands
  - Part 3: Transaction Processing
Divided into 3 documents

IC Card Specification

- Part 1: Electromechanical Characteristics, Logical Interface & Transmission Protocol
- Part 2: Data Elements & Commands
- Part 3: Application Selection
- Part 4: Security Aspects
IC Card Terminal Specification
- Part 1: General Requirements
- Part 2: Software Architecture
- Part 3: Cardholder, Attendant and Acquirer Interface IC Card Terminal Specification

IC Card Application Specification
EMV 2000

- Book 1: Application Independent ICC to Terminal Interface Requirement
- Book 2: Security and Key Management
- Book 3: Application Specification
- Book 4: Cardholder, Attendant and Acquirer Interface Requirements
Book 1: Application Independent ICC to Terminal Interface Requirement

- Part 1: Electromechanical characteristics, logical interfaces & transmission protocol equivalent to ISO-7816 parts 1, 2 and 3
- Part 2: File Commands and Application Selection
Book 2: Security & Key Management

- Static & Dynamic Authentication
- PIN Encipherment & Verification
- Application Cryptogram & Issuer Authentication
- Secured Messaging
- CA PK Management Principles & Policies
- Terminal Security & Key Management Requirements
Book 3: Application Specification

- Part 1: Data Elements & Commands
- Part 2: Debit and Credit Application Specification
  - Files for financial transaction interchange
  - Transaction flow
  - Generate AC coding
  - Functions used in transaction processing
Part 1: General Requirements
- Terminal types & capabilities
- Functional requirements
- Physical characteristics
- Security requirements

Part 2: Software Architecture

Part 3: Cardholder, Attendant and Acquirer Interface
## EMV Card

<table>
<thead>
<tr>
<th>Issuer Requirements</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>VIS 1.4</td>
<td>M/Chip</td>
</tr>
<tr>
<td>EMV Specs</td>
<td>ISO 7816</td>
</tr>
</tbody>
</table>
EMV Specifications: Objectives

- Universal Acceptance of Chip Debit / Credit Card
- Ensure that payment functions are performed consistently & securely at the point of transaction
- Define minimum functionalities to support International interoperability
EMV Concepts

- **Offline** risk management decision taken by:
  - Terminal (acquirer)
  - Card (issuer)

- 3 possible outcomes:
  - Offline approval of transaction
  - Online approval of transaction
  - Denial of transaction

- Card decision made according to Risk Management Rules defined by the issuer
EMV Concepts

- EMV is a “toolbox.”
- Each issuer is free to decide on the rules on:
  - Security
  - Risk management
  - Implementation
- Each acquirer is free to decide on his own risk management parameters.
1. I want to make a $20 transaction – OK?

2. Yes. This is the signature.

Risk Management Rules
Online if:
- Transaction > $40
- Every 3 offline transactions
EMV Concept:
Rejected Offline Example

1. I want to make a $30 offline transaction – OK?
2. No. Please ask my issuer bank.
5. Your bank says OK...
6. Here is your signature.
3. Is a $30 online transaction OK?
4. OK.

Risk Management Rules
Online if:
- Transaction > $40
- Every 3 offline transactions
EMV Concept:
Online Example

1. I want to make a $100 online transaction – OK?

2. Yes. Please ask my issuer bank.

3. Is a $100 online transaction OK?

4. OK.

5. Your bank says OK.

6. Here is your signature.

ISSUER

Risk Management Rules
Online if:
- Transaction > $40
- Every 3 offline transactions
Offline Transaction

- Authorization Controls
- Cardholder Verification
- Offline Data Authentication
- TC generation

Store → Acquirer → VisaNet → Issuer

TC 05

- Transaction Certificate
- New Data
Online Transaction

- Authorization Controls
- Cardholder Verification
- Offline Data Authentication
- ARQC generation (for online request) ★
- ARPC validation (for online response) ★
- TC generation (for clearing)

- Authorization Request Cryptogram
- New Data/Results of offline risk management
- Authorization Response Cryptogram
- New Data
- Post-Issuance Updates

- TC 05
- Transaction Certificate
- New Data
EMV Specification Coverage

- Data Authorization
- Data Collection

- Transaction Storage
- Communication Flow & Data

Not specified by EMV Specification

EMV Specification

- Risk Management
- Cryptography of transaction
- Personalization

Not specified by EMV Specification
Terminal Coverage

Europay, Mastercard or Visa Requirements

- Acquirer Parameters
- Transaction Log
- Communication Protocol

- Parameters
- Authorization Format
- Specific Functions

- Transaction Sequence
- Interface

EMV Terminal Application

EMV
EMV Terminal Transaction Flow

1. Application Selection
2. Initiate Application
3. Read Application Data
4. Static/Dynamic Data Authentication
5. Processing Restriction
6. Cardholder Verification
7. Terminal Action Analysis
8. Card Action Analysis
9. Completion
10. Online Processing & Issuer Authentication
11. Script Processing

Decision:
- On/Offline?
  - Online
  - Offline

Terminal Risk Management
Transaction Functional Blocks

- Application Selection
- Card Authentication
- Cardholder Identification
- Authorization / Acceptance of Transaction
- Script Processing
Application Selection

1. What applications do you have?

2. I have the EMV D/C application and the X Private Application

3. I only know the EMV D/C application and I select the EMV D/C application.
1. What applications do you have?

2. I have the EMV D/C application and the E-Purse Application

3. I know both but you have priority over the EMV D/C application and therefore I select the EMV D/C application.
1. What applications do you have?

2. I have the EMV D/C application and the Y Private Application

3. I know both applications but the cardholder has chosen EMV D/C, therefore I select EMV D/C.
EMV Card Capabilities

- Authorization Controls
- Cardholder Verification Methods
- Usage Controls
- Authentication
- Dynamic Data Updates
- Exception Handling
- Multiple Functions
Authorization Controls

- Issuer-defined authorization parameters are based on the risk associated with the transaction type or the POS environment (e.g. online authorization, purchase limit and offline transaction counters).
- The chip authorizes the offline transaction.
- Offline authorization reduces fraud and lower costs.
- The issuer may establish default online / offline modes depending on the product or account.
- The offline default may trigger the online mode based on:
  - Reaching a preset limit to the offline activity (time / amount limit)
  - First time use
  - Type of transaction (e.g. cashback)
  - Conditions at the POS (e.g. PINpad failure)
Usage Controls

The chip manages card use based on conditions at the point of transaction using parameters and the processing power of the chip.

- **Geographic**
  - Restrict to domestic / international

- **Transaction**
  - Restrict usage to local goods & services, ATM, goods & services for international transactions, etc.

- **Inactive or expired accounts**
  - Force online or decline transaction

- **Ceiling value of cash or cashback transaction**

- **Maximum transaction amounts allowed**

- **Restricted usage based on merchant type or terminal type**
Authentication

The chip enables a set of risk management tools, which combat fraud involving cryptology & logical comparison between the transaction and card data, to verify the legitimacy of the card and the host.

- Offline data authentication to prevent fraudulent or altered data
- Online card authentication to detect counterfeited card
- Issuer authentication for dynamic data update
- Transaction certificate to provide information confirming that actual steps and processes are performed by the card, the terminal and the merchant during a given transaction.

*Risk management tools control fraud & provide information that ensure integrity of card transactions.*
Dynamic Data Update

Data inside the chip can be updated at the POS without reissuing the card, thus providing convenience to both the cardholder and issuer, and enhancing risk control.

- Blocking an application or the entire card
- Unblocking an application or the entire card
- Resetting the PIN-try counter
- Changing the upper consecutive offline limit
- Changing the lower consecutive offline limit
Exception Handling

- **On-line inoperative**
  - The issuer can designate in the card a maximum number of offline transactions when the online processing is no longer operative.

- **PIN-try limit exceeded**
  - The issuer has the ability to allow more tries under certain circumstances.

- **Terminal fault**
  - Merchants can accept transactions using magnetic stripe.

- **Network fault**
  - The processor is allowed to edit the transaction.

*This feature provides issuers with greater flexibility to customize payment services on the basis of their risk assessment.*
Multiple Function

- The chip can store information about multiple functions.
- The chip can communicate with various devices to allow the selection of different applications at the point of transaction.
- The magnetic stripe can provide access to other services. (eg. ATM)
- It is possible to use the chip for other applications. (eg. loyalty, electronic purse, membership card, etc.)
EMV Terminal Transaction Flow

1. Application Selection
2. Initiate Application
3. Read Application Data
4. Static / Dynamic Data Authentication
5. Processing Restriction
6. Cardholder Verification
7. Terminal Action Analysis
8. Card Action Analysis
9. On / Offline?
   - Offline
     - Completion
     - Script Processing
   - Online
     - Online Processing & Issuer Authentication
     - Terminal Risk Management
Card Authentication Issues

- Problem of an international environment (e.g., problem of sharing secrets)
- Authenticating a Taiwanese EMV card, for instance, in a Japanese terminal
- No direct link established between the card issuer and the terminal application
- Public key cryptography as a solution to this problem
Public key cryptography (RSA) requires a public/private key pair to be generated by the payment scheme and the issuer.

The owner of the private key is the only one who can sign the message.

The public key is known to everyone, and hence able to authenticate the author of the message.
Static Data Authentication (SDA)

- **Issuer**
  - Private Key: $Si(issuer)$
  - Public Key: $Pi(issuer)$
  - $Pi$ certified with $Ss$

- **Payment Scheme**
  - Private Key: $Ss(scheme)$
  - Public Key: $Ps(scheme)$

- **Acquirer**
  - Distributed to the acquirer, stored in the terminal

$Si$ used to create the digital signature of the card
Step 1: Certification of Pi

- The issuer certifies the card.
- The issuer stores the result (SDA) in the card.
Step 2: Personalization

- The payment organization acts as a certification authority.
- The payment organization’s Ss is used to certify the issuer’s Pi.

Distributed to the acquirer, stored in the terminal.
Step 3: Authentication

- The terminal verifies the Pi certificate to ensure issuer authenticity.
- The terminal then verifies the data certificate to ensure card authenticity.
- The proof of card authenticity is **static**.

![Diagram showing the process of authentication between Issuer, Payment Scheme, and Acquirer]

- **Issuer**
  - Private Key: Si(issuer)
  - Public Key: Pi(issuer)

- **Payment Scheme**
  - Private Key: Ss(scheme)
  - Public Key: Ps(scheme)

- **Acquirer**

Pi certified with Ss

Distributed to the acquirer, stored in the terminal

Si used to create the digital signature of the card
Dynamic Data Authentication (DDA)

- Payment organization certifies the issuer
- The issuer certifies the card
- The card **dynamically** proves its authenticity to the terminal
Card Authentication

- SDA is the storage of:
  - A certificate for issuer authentication
  - A digital signature for card authentication

- DDA is storage of:
  - A certificate for issuer authentication
  - A certificate for card authentication

- Dynamic generation of signature for authentication
DDA

- RSA calculation needs a smart card with a cryptographic co-processor.
- The time it takes to produce a digital signature (1024 bits) is approximately 800 ms.
The issuer defines a method and the conditions for identification.
The terminal executes according to the agreed methods and conditions.
Confirmation of holder identity (photo) & acceptance of the transaction (signature panel) occurs.
Offline PIN verification is done by comparing it with the PIN in the chip.
Encrypted online PIN verification is done by the host.
The PIN is optional and dependent on issuer market requirements, merchant segments and terminal types.
The chip stores and processes issuer instructions on which CVMs are to be used in different situations.
This process enhances security and improves issuer control.
Cardholder Verification Example

- PIN online if cash, else
- PIN offline if < $50, else
- Signature

- PIN Online if the transaction is cash
- Otherwise, PIN offline if the transaction < $50
- Otherwise, paper signature if the PIN offline is incorrect
Offline Cardholder Identification

**Plain**
- Verify PIN 1233
- PIN OK!

**Ciphered**
- Verify PIN “#$&@”
- PIN OK!

- Ciphering 1234 = #$&@

PIN 1234 = Card PIN?

Deciphering #$&@
- Equals Card PIN?
EMV Terminal Transaction Flow

1. Application Selection
2. Initiate Application
3. Read Application Data
4. Static / Dynamic Data Authentication
5. Processing Restriction
6. Cardholder Verification
7. Terminal Action Analysis
8. Card Action Analysis
9. Completion
10. On / Offline?
   - Online
   - Offline
11. Online Processing & Issuer Authentication
12. Script Processing
Transaction Authorization/Validation

- Acquirer Risk Management
- Terminal’s Decision
- Card’s Decision
- Issuer Risk Management
Terminal risk management is defined by the acquirer.

It consists of:

- Checking floor limit: compare with the transaction amount
- Random transaction selection: to perform transaction online
- Velocity checking: after a number of consecutive offline transactions, the transaction should go online depending on consecutive limits, cumulative total, international limits, dual currency amount and limits
Terminal Decision

- Analyze the result of previous functions.
  - Card authentication result
  - Cardholder identification result
  - Acquirer risk management result
- Based on the result, a joint acquirer-issuer decision is made.
Terminal Action Analysis

I reject the transaction.

I want an online transaction.

I want an offline transaction.

Application Authentication Cryptogram (AAC)

Authorization Request Cryptogram (ARC)

Transaction Certificate (TC)
EMV Terminal Transaction Flow

1. Application Selection
2. Initiate Application
3. Read Application Data
4. Static / Dynamic Data Authentication
5. Processing Restriction
6. Cardholder Verification
7. Terminal Action Analysis
8. Card Action Analysis
9. On / Offline?
10. Offline
11. Completion
12. Online
13. Online Processing & Issuer Authentication
14. Script Processing
The card performs its own risk management (not specified by EMV) and makes the final decision.
Issuer Risk Management

- Performed by the Generate AC Command
- Issuer decides its own rules
- Examples of possible rules:
  - Counting total consecutive number of offline transactions
  - Counting total consecutive amount of offline transactions
  - Incorrect identification of cardholder
  - Verification of previous transaction
  - And more...

Generate AC

Online, Offline, or Rejected

EMV

Not specified by EMV
Script Processing Mechanism

Card Issuing Bank 1

ISSUER

Bank 2 Country 2 Terminal

Direct link between the card & the issuer

Bank 1 Country 1 Card

Risk Management Rules
Online if:
- Transaction > $40
- Every 3 offline transactions
Script Processing Mechanism

- Allows issuer to be in contact with their cards during online transactions
- Independence of country and acquirer
- To do what?
  - Change card parameters
  - Blocking and unblocking of application
  - And more...
EMV Card Application

- IAC
- Card Risk Management
- Cryptography
- Personalisation

Europay, Mastercard or Visa requirements

EMV
ICC Specifications for Payment Systems

- Part 1: Electromechanical Characteristics, Logical Interface and Transmission Protocols
- Part 2: Data Elements and Commands
- Part 3: Application Selection
- Part 4: Security Aspects
EMV ‘96

- ICC Terminal Specification for Payment Systems
  - Part 1: General Requirements
  - Part 2: Software Architecture
  - Part 3: Cardholder, Attendant & Acquirer Interface
- ICC Application Specification for Payment Systems
Card Specification

Prerequisite documents to understand Part 1: ISO-7816 1,2,3

- Essentially the EMV implementation of ISO-7816 parts 1,2,3
- Defines Answer To Reset (ATR) characters
- Requires a warm reset if ATR is different
  - Possible migration from proprietary/national system to co-exist with EMV without modification of existing systems (eg. Taiwan FISC, Singapore CashCard, etc.)
- Allows the card to support either the T=0 or T=1 protocol
- Does not require Vpp
Prerequisite document to understand Part 2: ISO-7816-4,6

- Defines all data objects (more than 100)
- Data objects are in TLV format
- Can be a primitive data object (eg. TLV or a constructed data object like TL(TLV)..(TLV))
- Defines the range of the SFI (file name) to be used
- Defines the EMV command set
- And more...
EMV Card Commands

- 8x 1E: Application Block
- 8x 18: Application Unblock
- 8x 16: Card Block
- 0x 82: External Authentication
- 8x AE: Generate Application Cryptogram
- 0x 84: Get Challenge (added in EMV2000)
- 8x CA: Get Data
- 8x A8: Get Processing Options
- 0x 88: Internal Authentication
- 8x 24: PIN Change / Unblock
- 0x B2: Read Record
- 0x A4: Select
- 0x 20: Verify
- 8xDx, 8xEx, 9xxx, Exxx: Reserved
Application Selection

- Terminal cold reset card; If not an EMV card, warm reset
- SELECT PSE DDF name = 1PAY.SYS.DDF01
- Read FCI using Get Response
- Read DIR EF SFI using READ RECORD
- Read supported applications using READ RECORD & match supported applications
- Select the highest priority application supported by the terminal using the SELECT command on the ADF
Data Elements for Financial Transaction

- ICC data objects are stored in:
  - Fixed sized records
  - Variable size records
- All objects are in TLV format.

**Primitive data object:**

```
Tag (1 or 2 bytes) length (1 byte) value
```

**Constructed data object:**

```
TL (TLV)(TLV) . . . (TLV)
```

*The value field of a constructed object (TLV)(TLV) . . . (TLV) is called a template.*
Tag Structure

- Primitive object tag – 0x, 4x, 5x, 8x, 9x, Cx, Dx
- Constructed object tag – 2x, 3x, 6x, 7x, Ax, Bx, Ex, Fx
- 2 bytes tag – odd F (eg. 7Fxx, 9Fxx)
- Tag is always within the range of 1F to 7F
## Examples of Data Objects

<table>
<thead>
<tr>
<th>Tag</th>
<th>Length</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>5F24</td>
<td>3</td>
<td>Application Expiry Date</td>
</tr>
<tr>
<td>5A</td>
<td>10</td>
<td>Application Primary Account Number</td>
</tr>
<tr>
<td>8C</td>
<td>Variable</td>
<td>Card Risk Management Data</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Object List 1</td>
</tr>
<tr>
<td>8D</td>
<td>Variable</td>
<td>Card Risk Management Data</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Object List 2</td>
</tr>
</tbody>
</table>

*The above are mandatory data objects.*
### Examples of Data Objects

<table>
<thead>
<tr>
<th>Tag</th>
<th>Length</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>8F</td>
<td>1</td>
<td>Certification Authority Public Key Index</td>
</tr>
<tr>
<td>90</td>
<td>40-128</td>
<td>Issuer Public Key Certificate</td>
</tr>
<tr>
<td>93</td>
<td>40-128</td>
<td>Signed Application Data</td>
</tr>
<tr>
<td>92</td>
<td>1-34</td>
<td>Issuer Public Key (Remainder)</td>
</tr>
<tr>
<td>9F32</td>
<td>1-32</td>
<td>Issuer Public Key (Exponent)</td>
</tr>
</tbody>
</table>

The above are static data authentication data objects.
Record Data Object

- Record in SFI 1 – 10 must be in BER-TLV.
  - SFI 1 to 10 is governed by the EMV specification.
  - SFI 11 to 20 is proprietary data of payment systems.
  - SFI 21 to 30 is proprietary data of the issuer.
- The tag of a record data object is 70, indicating that it is a constructed data object.
- The Application File Locator (AFL) indicates files & records used for transaction processing.
Data Object Existence

- **M** = Mandatory, must be present to allow terminal transaction processing
- **R** = Required, terminal should not terminate transaction if not received
- **C** = Conditional, necessary under certain conditions
- **O** = Optional, necessary under certain conditions

*R and C are defined by Visa in the VSDC Requirements for Common Personalization document*
IC Card Structure

- **MF**: Master File, equivalent to root directory
- **DF**: Dedicated File, equivalent to sub-directory
- **EF**: Elementary File, equivalent to a data file; also called AEF or Application EF
- **DIR File**: EF containing a list of applications supported by the card
- **DDF**: Directory Definition File
- **ADF**: Application Definition File, contains a list of AEF's for this application
EMV Card Organization

- **PSE** is usually also the MF.

- **DDF**
  - 1PAY.SYS.DDF01
  - FCI of PSE
  - VSDC ADF A000000003000

- **EPURSE ADF ADF_NAME**
  - DIR EF
    - SFI 01
  - FCI of DIR EF

- **Cardholder Data EF,01**
- **Application Data EF,02**
- **Authentication Data EF,03**
- Etc.
DIR EF

- An EF residing inside a DDF
- SFI must be between 1 to 10, where value is in DDF FCI
- Used in the Application Selection process
- Each entry is a record of the Application Template (tag 61)
- An Application Template is a constructed object

<table>
<thead>
<tr>
<th>Tag</th>
<th>Length</th>
<th>Value</th>
<th>Presence</th>
</tr>
</thead>
<tbody>
<tr>
<td>4F</td>
<td>5-16</td>
<td>ADF name (AID)</td>
<td>Mandatory</td>
</tr>
<tr>
<td>50</td>
<td>1-16</td>
<td>Application Label</td>
<td>Mandatory</td>
</tr>
<tr>
<td>9F12</td>
<td>1-16</td>
<td>Application Preferred Name</td>
<td>Optional</td>
</tr>
<tr>
<td>87</td>
<td>1</td>
<td>Application Priority</td>
<td>Optional</td>
</tr>
</tbody>
</table>
### DIR EF Content

<table>
<thead>
<tr>
<th>Code</th>
<th>Offset</th>
<th>Description</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>70h</td>
<td>15h to 2Fh</td>
<td>61h</td>
<td>13h to 2Dh</td>
</tr>
<tr>
<td>50h</td>
<td>01h to 10h</td>
<td>Application Label, up to 16 bytes (M)</td>
<td></td>
</tr>
<tr>
<td>9F12h</td>
<td>01h to 10h</td>
<td>Application Preferred Name, (O) Up to 16 bytes</td>
<td></td>
</tr>
<tr>
<td>87h</td>
<td>01h</td>
<td>Application Priority Indicator, 1 byte (O)</td>
<td></td>
</tr>
<tr>
<td>73h</td>
<td>04h</td>
<td>CEh</td>
<td>02h</td>
</tr>
</tbody>
</table>

*70h = template proprietary to this application
61h = application template*
DDF

- Implemented as a DF inside the card
- Mandatory to have the 1PAY.SYS.DDF01 DDF, the Payment System Environment (PSE)
- Get Response after Select DDF returns the FCI template
- A template is a constructed data object
- One implementation of FCI uses a transparent EF inside the DF to store the Get Response content
<table>
<thead>
<tr>
<th>Tag</th>
<th>Value</th>
<th>Presence</th>
</tr>
</thead>
<tbody>
<tr>
<td>6F</td>
<td>FCI Template</td>
<td>Mandatory</td>
</tr>
<tr>
<td>84</td>
<td>DF Name</td>
<td>Mandatory</td>
</tr>
<tr>
<td>A5</td>
<td>FCI Proprietary Template</td>
<td>Mandatory</td>
</tr>
<tr>
<td>88</td>
<td>SFI of DIR EF</td>
<td>Mandatory</td>
</tr>
<tr>
<td>5F2D</td>
<td>Language Preference</td>
<td>Optional</td>
</tr>
<tr>
<td>9F11</td>
<td>Issuer Code Table Index</td>
<td>Optional</td>
</tr>
<tr>
<td>9F3B</td>
<td>Application Reference Currency</td>
<td>Optional</td>
</tr>
<tr>
<td>BF0C</td>
<td>FCI Issuer Discretionary Data</td>
<td>Optional</td>
</tr>
<tr>
<td>XXXX</td>
<td>Additional object as in book 3</td>
<td>Optional</td>
</tr>
<tr>
<td>Tag</td>
<td>Value</td>
<td>Presence</td>
</tr>
<tr>
<td>-----</td>
<td>--------------------------------------</td>
<td>------------</td>
</tr>
<tr>
<td>6F</td>
<td>FCI Template</td>
<td>Mandatory</td>
</tr>
<tr>
<td>84</td>
<td>DF Name</td>
<td>Mandatory</td>
</tr>
<tr>
<td>A5</td>
<td>FCI Proprietary Template</td>
<td>Mandatory</td>
</tr>
<tr>
<td>88</td>
<td>SFI of DIR EF</td>
<td>Mandatory</td>
</tr>
<tr>
<td>BF0C</td>
<td>FCI Issuer Discretionary Data</td>
<td>Optional</td>
</tr>
<tr>
<td>XXXX</td>
<td>Additional object as in book</td>
<td>Optional</td>
</tr>
</tbody>
</table>
ADF

- Implemented as a DF inside the card
- Can have many ADFs, with one ADF per application (eg. Credit Card, Electronic Purse, etc.)
- Each ADF entry can be found in the DIR EF of the 1PAY.SYS.DDF01
- The Application Priority byte indicates:
  - Application selected without holder's confirmation
  - Application selected with holder's confirmation
  - The priority of the selection from 1(highest) to 15 or none
  - Catered for RFU (reserved for future use)
<table>
<thead>
<tr>
<th>Tag</th>
<th>Value</th>
<th>Presence</th>
</tr>
</thead>
<tbody>
<tr>
<td>6F</td>
<td>FCI Template</td>
<td>Mandatory</td>
</tr>
<tr>
<td>84</td>
<td>DF Name</td>
<td>Mandatory</td>
</tr>
<tr>
<td>A5</td>
<td>FCI Proprietary Template</td>
<td>Mandatory</td>
</tr>
<tr>
<td>87</td>
<td>Application Priority Indicator</td>
<td>Mandatory</td>
</tr>
<tr>
<td>9F38</td>
<td>Processing Options Data Object List</td>
<td>Optional</td>
</tr>
<tr>
<td>BF0C</td>
<td>FCI Issuer Discretionary Data</td>
<td>Optional</td>
</tr>
</tbody>
</table>
EMV specifications only define:
- The structure
- The commands to access files
- Data objects

The issuer will map the appropriate data objects to files (SFI 1-10) according to their needs, BUT in compliance with the rules
- Linear FREE READ, but may be a conditional UPDATE
- Each record is limited to 254 bytes, including tag & length
- Each record is a constructed data object, tag 70

AFL defines the file & record required for application processing, a response from Get Processing Options.
## Cardholder-related Data File

<table>
<thead>
<tr>
<th>Tag</th>
<th>Value</th>
<th>Presence</th>
</tr>
</thead>
<tbody>
<tr>
<td>5F24</td>
<td>Application Expiry Date</td>
<td>M</td>
</tr>
<tr>
<td>5A</td>
<td>Application PAN</td>
<td>M</td>
</tr>
<tr>
<td>5F25</td>
<td>Application Effective Date</td>
<td>O</td>
</tr>
<tr>
<td>5F34</td>
<td>Application PAN Sequence Number</td>
<td>O</td>
</tr>
<tr>
<td>5F20</td>
<td>Cardholder Name</td>
<td>O</td>
</tr>
<tr>
<td>9F0B</td>
<td>Cardholder Name Extended</td>
<td>O</td>
</tr>
<tr>
<td>5F28</td>
<td>Issuer Country Code</td>
<td>O</td>
</tr>
<tr>
<td>5F30</td>
<td>Service Code</td>
<td>O</td>
</tr>
<tr>
<td>9F1F</td>
<td>Track 1 Discretionary Data</td>
<td>O</td>
</tr>
<tr>
<td>57</td>
<td>Track 2 Equivalent Data</td>
<td>O</td>
</tr>
<tr>
<td>9F20</td>
<td>Track 2 Discretionary Data</td>
<td>O</td>
</tr>
</tbody>
</table>
## Application-related Data File

<table>
<thead>
<tr>
<th>Tag</th>
<th>Value</th>
<th>Presence</th>
</tr>
</thead>
<tbody>
<tr>
<td>8C</td>
<td>Card Risk Management Data Object List1</td>
<td>M</td>
</tr>
<tr>
<td>8D</td>
<td>Card Risk Management Data Object List2</td>
<td>M</td>
</tr>
<tr>
<td>9F05</td>
<td>Application Discretionary Data</td>
<td>O</td>
</tr>
<tr>
<td>9F07</td>
<td>Application Usage Control</td>
<td>O</td>
</tr>
<tr>
<td>9F08</td>
<td>Application Version Number</td>
<td>O</td>
</tr>
<tr>
<td>9F14</td>
<td>Lower Consecutive Offline Limit</td>
<td>O</td>
</tr>
<tr>
<td>9F23</td>
<td>Upper Consecutive Offline Limit</td>
<td>O</td>
</tr>
<tr>
<td>8E</td>
<td>Cardholder Verification Method List</td>
<td>O</td>
</tr>
<tr>
<td>97</td>
<td>Transaction Certificate DOL</td>
<td>O</td>
</tr>
<tr>
<td>9F0D</td>
<td>Issuer Action Code - Default</td>
<td>O</td>
</tr>
<tr>
<td>9F0E</td>
<td>Issuer Action Code - Denial</td>
<td>O</td>
</tr>
<tr>
<td>9F0F</td>
<td>Issuer Action Code - Online</td>
<td>O</td>
</tr>
</tbody>
</table>
## Application-related Data File

<table>
<thead>
<tr>
<th>Tag</th>
<th>Value</th>
<th>Presence</th>
</tr>
</thead>
<tbody>
<tr>
<td>9F42</td>
<td>Application Currency Code</td>
<td>O</td>
</tr>
<tr>
<td>9F44</td>
<td>Application Currency Component</td>
<td>O</td>
</tr>
<tr>
<td>9F4A</td>
<td>Static Data Authentication Tag List</td>
<td>O</td>
</tr>
<tr>
<td>Tag</td>
<td>Value</td>
<td>Presence</td>
</tr>
<tr>
<td>-----</td>
<td>--------------------------------------------------------------</td>
<td>----------</td>
</tr>
<tr>
<td>8F</td>
<td>Certification Authority Public Key Index</td>
<td>M</td>
</tr>
<tr>
<td>90</td>
<td>Certified Issuer Public Key</td>
<td>M</td>
</tr>
<tr>
<td>93</td>
<td>Signed Application Data</td>
<td>M</td>
</tr>
<tr>
<td>92</td>
<td>Issuer Public Key Index</td>
<td>O</td>
</tr>
</tbody>
</table>
Quick, easy and cost-effective implementation of ICC programs

Infra-structure supporting future ICC products
  - Multiple applications
  - Global interoperability
  - Co-existence with non-Visa programs
  - Avoidance of confusion

And more...
Complies with ICC Specification Part 1
Complies with ICC Specification Part 3 - Application Selection
Supports a card file with 1 record
  - Track 2 data
  - Track 1 - Cardholder name and track 1 discretionary data
Complies with the EMV data coding scheme
Processes transactions using a message format identical to the current magnetic transaction
No longer in use
VSDC Card

- Can be a native card (e.g., conventional chip operating system powered type of card)
- Can also be a Global Platform Java Card
Data Preparation before Personalization

- Issuer public key certificate, remainder, exponent
- Signed static application data
- Uniquely Derived Key (derived from PAN and protected by Key for card authentication)
- MAC Derived Key
- ENC Derived Key
- Offline PIN
3 Alternatives for VSDC

- Quick Start data elements in VSDC
- Jump Start data elements in VSDC
- Full data elements in VSDC
What QuickStart Cannot Do

- Velocity Checking
- Static Data Authentication
- Dynamic Data Authentication
- Script Processing
- Offline PIN
- Offline
What Jump Start **Cannot Do**

- Velocity Checking
- Dynamic Data Authentication
- Script Processing
- Offline PIN
- Offline
Data Elements in VSDC

- Magnetic Stripe Image (MSI)
- Authorization Control (AuthC)
- Static Data Authentication (SDA)
- Dynamic Data Authentication (DDA)
- Online Card / Issuer Authentication (CAM / IAuth)
- Issuer Script for Post Issuance Update (IS)
Possible VSDC Templates

Template

1. Magnetic Stripe Image
2. Authorization Control
3. Enhanced Cardholder Verification Method (PIN)
4. Offline Static Data Authentication (SDA)
5. Online Card and Issuer Authentication (CAM)
6. SDA, Offline PIN, and Authorization Controls
7. SDA, Offline PIN and CAM
8. SDA, Offline PIN, CAM, and Authorization Controls
9. SDA, CAM, and Authorization Controls
10. Offline Dynamic Data Authentication (DDA)
11. Post Issuance Updates (Issuer Script – IS)
Application Interchange Profile (AIP)

- Card indicates processing capabilities
- Returns via Get Response after Get Processing Options APDU
- Returned data – domestic & international AIP, AFL:
  - Tag 80, L=var AIP (2 bytes) AFL(n*4bytes)
  - Tag 80, L=var AIP (2 bytes) AFL (n*4bytes)

AIP Definition:

<table>
<thead>
<tr>
<th>Byte 2 Bit 4</th>
<th>Terminal Risk Management</th>
</tr>
</thead>
<tbody>
<tr>
<td>Byte 2 Bit 3</td>
<td>Issuer Authentication</td>
</tr>
<tr>
<td>Byte 2 Bit 2,1</td>
<td>RFU</td>
</tr>
<tr>
<td>Byte 1, Bit 8-1</td>
<td>RFU</td>
</tr>
</tbody>
</table>
## VSDC Template & AIP

<table>
<thead>
<tr>
<th>VSDC Template</th>
<th>X</th>
<th>SDA</th>
<th>DDA</th>
<th>Cardholder Verification</th>
<th>Terminal_Risk_Management</th>
<th>Issuer_Authentication</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. MSI</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>00 000000000</td>
</tr>
<tr>
<td>2. Authorization Control</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>00 000000000</td>
</tr>
<tr>
<td>3. Enhanced CVM (PIN)</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>00 000000000</td>
</tr>
<tr>
<td>4. SDA</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>00 000000000</td>
</tr>
<tr>
<td>5. Online Card &amp; Issuer Auth (CAM)</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>10 000000000</td>
</tr>
<tr>
<td>6. SDA, Offline PIN &amp; Authorization</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>00 000000000</td>
</tr>
<tr>
<td>7. SDA, Offline &amp; CAM</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>10 000000000</td>
</tr>
<tr>
<td>8. SDA, Offline, CAM &amp; Auth Control</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>10 000000000</td>
</tr>
<tr>
<td>9. SDA, CAM &amp; Auth Control</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>10 000000000</td>
</tr>
<tr>
<td>10. DDA</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>00 000000000</td>
</tr>
<tr>
<td>11. Post Issuance Update (IS)</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>10 000000000</td>
</tr>
</tbody>
</table>
Each AFL is a 4 byte-pointer

- First byte – SFI
- Second byte – record # of first record (r1) to be read
- Third byte – record # of last record (r2) to be read
- Fourth byte – number of consecutive records involved in the SDA starting from r1
Questions?