



Chip and PIN Programme
Guideline G18
Configuring Integrated Systems

The information contained within this document has been prepared by the Chip and PIN PMO, for use by participants in the Programme only. Participants rely on the information contained herein at their own risk. Other third parties should not rely on the information.



TABLE OF CONTENTS

1	INTRODUCTION.....	4
2	WHAT ARE THE ISSUES?	4
3	SCOPE.....	5
4	REFERENCE DOCUMENTS	5
5	GLOSSARY OF TERMS	6
6	HARDWARE	7
6.1	Hardware configuration at Point of Sale	7
6.2	Card reader / writer.....	7
6.3	PINpads	8
6.4	Point of sale terminal	9
6.5	Other hardware components.....	9
6.6	Accessibility	9
7	SOFTWARE.....	9
7.1	Architecture.....	9
7.2	Parameters.....	11
7.2.1	Floor limits and associated parameters.....	12
7.2.2	Terminal Type	12
7.2.3	Terminal Capabilities.....	12
7.2.4	Additional Terminal Capabilities	13
7.2.5	Other ICS options	14
8	TESTING REQUIREMENTS	16
8.1	Pre-Certification Testing.....	16
8.2	EMV Level 1	16
8.3	EMV Level 2	16
8.4	Application Functional Testing	17
8.5	Bank Accreditation.....	17
8.6	Test Tools	18
8.7	Common Problems Encountered.....	18
8.8	Recertification.....	19
8.8.1	EMV Level 1	19
8.8.2	EMV Level 2	19
8.8.3	PINpad security approval	20



Guideline G18 – Configuring Integrated Systems

8.8.4 Bank accreditation.....20

9 CONCLUSIONS.....20

APPENDIX A.....22

FUNCTIONAL SERVICES DESCRIBED IN BRC CHIP ARCHITECTURE22



1 Introduction

The purpose of this document is to assist vendors to design and implement software solutions for distributed retail systems in a way that both meets the requirements of the card schemes and acquirers, and allows systems to be configured with a minimum of recertification.

The document is required because vendors have experienced delays and extra costs at the certification stage (both EMVCo and Acquirer Acceptance Testing); these not only lead to corresponding delays and costs of their current retailer implementations, jeopardising the Programme plans in the short term, but also deter further retailers from implementing, which will jeopardise longer-term plans.

Although there is work under way within EMVCo to help resolve some of the issues described (in particular a proposal for testing of multiple configurations) vendors need guidance now to minimise the effect on the Chip and PIN Programme.

The document requires knowledge of smart card and point of sale technology, as well as some familiarity with the structure and terminology of EMV.

2 What are the issues?

Many medium to large stores utilise integrated and distributed till systems to perform point of sale transactions. A payment card transaction may form only a relatively small part of the overall configuration.

EMV introduces the concept of a terminal to perform the card transaction functions. Such terminals are required to undergo extensive type approval processes (see Programme Guideline G6). However, with integrated systems, the functions of the 'terminal' may be spread across a number of components and locations. In particular, this document is concerned with the case where some EMV functions (for example, hotlist checking as part of Terminal Risk Management) are performed in a central system that controls multiple tills.

Changes to hardware and software components (both within and outside the EMV kernel) may impact part of the terminal approval process. Vendors and retailers often consider these changes to be minor configuration or parameter changes, but they may result in time consuming and expensive re-certification processes. The selection of the correct architecture for such systems is critical to ensure that they remain flexible, efficient and cost-effective.

There is no perfect solution: systems design involves compromises between performance, cost and flexibility, between ease of initial certification and the extent to which future changes can be made without further certification. Vendors should make conscious decisions as to the optimum balance for their systems: see Figure 1.1 below.

In order to provide this flexibility, as well as the key management that EMV demands, some form of Terminal Management System is usually necessary, to manage the PINpads or other intelligent devices at the Point of Sale. This is usual for stand-alone terminals but not normal for integrated systems accepting only magstripe transactions. Vendors and retailers must understand the requirements for such systems.

Guideline G18 – Configuring Integrated Systems

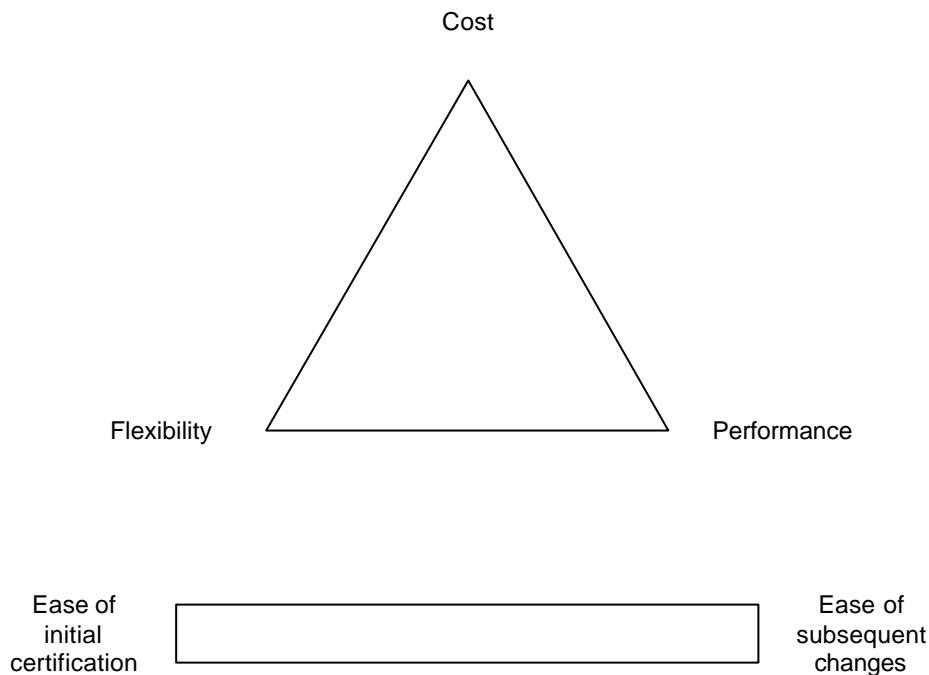


Figure 1.1 - Balancing requirements

3 Scope

This document provides guidance on the way that software solutions and systems should be set up, where parameters should be stored, and how they should be presented for type approval and testing, in order to ensure that the software can be used in all likely and planned configurations. This includes:

- a detailed description of the conditions under which recertification is required, by EMV, card schemes and acquirers;
- guidance on the range of parameters likely to be required in UK merchant systems, both face-to-face and unattended;
- guidance as to which functions must form part of the kernel, and which should normally be kept outside of the kernel;
- guidance as to the way parameters should be stored, and on the requirements for controlling parameter changes;
- guidance on the requirements for test harnesses and other conditions for submission for EMV, card schemes and acquirer testing;
- further comments on the requirements for Pin Entry Device testing.

4 Reference Documents

- **[APACS70]** *Card Acceptor To Acquirer Interface Standards*, Association for Payment & Clearing Services, 1st March 2004. This standard supersedes all the Chip and PIN Programme Recommendations in relation to Point of Sale systems and procedures.



Guideline G18 – Configuring Integrated Systems

- **[BRC]** *An integrated retail systems architecture for acceptance of chip cards: functional descriptions and interface guidelines*; Teconomica Ltd for the British Retail Consortium. V2.0, 15 July 1999. Contact brc@teconomica.co.uk
- **[G6]** Programme Guideline G6 – *End-to-end Certification Process for POS Equipment*. See www.chipandpin.co.uk.
- **[G12]** Programme Guideline G12 – *Accessibility*. See www.chipandpin.co.uk.
- **[EMV]** EMV Specifications (currently V4.1 dated May 2004) - www.emvco.com
- **[MCFA]** *M/Chip Functional Architecture*. MasterCard International, January 2003 (to be revised January 2005)
- **[PCI]** *Payment Card Industry PED Requirements V1.0*, Visa and MasterCard, April 2004
- **[PP]** APACS - PIN Entry Device Protection Profile Version 1.37 - 11th July 2003
http://www.cesg.gov.uk/site/iacs/itsec/media/protection-profiles/PED_PPv1_37.pdf
- **[SP3]** *Smart Payment Product Principles*. Visa EU V3.0, July 2003
http://www.chipandpin.co.uk/reflib/SP3_main_brochure.pdf
- **[TA]** EMV Type Approval requirements – www.emvco.com
- **[TA11]** **Type Approval Bulletin 11**: Major and minor changes – www.emvco.com
- **[VIS]** *Visa Integrated Circuit Card Application Overview*, Visa International V1.4.0, September 2001 - <http://international.visa.com/fb/paytech/smartcard/vsmartspecs/visspec.jsp>

5 Glossary of Terms

AID	Application Identifier: used in both ISO 7816 and EMV to identify applications on a smart card
API	Application Program Interface: a closely defined interface to which a program works
EMV	Integrated Circuit Card Specifications for Payment Systems, developed by Europay, MasterCard and Visa
EMVCo	Independent company incorporated to own and maintain the EMV Specifications
IAC	Issuer Action Code: a fixed set of parameters stored on the card and used by the issuer to control card behaviour
ICS	Implementation Conformance Statement: the list of functions and capabilities that must be submitted with a kernel or device when seeking EMVCo Level 2 approval
IFD	Interface Device: the card reader/writer for the chip card. (Also sometimes IFM: Interface Module)
PAL	Programmable Array Logic: a form of firmware often used to control hardware devices
PED	PIN Entry Device: a device comprising a PINpad, display and logic for checking and encrypting PINs.
POS	Point of Sale (or Point of Service)



Guideline G18 – Configuring Integrated Systems

PINpad	Keypad into which the customer types his or her PIN
TAC	Terminal Action Code: a set of parameters stored in the terminal and used by the acquirer to control terminal behaviour

6 Hardware

6.1 Hardware configuration at Point of Sale

The EMV specifications [EMV] were clearly written with a terminal implementation in mind. They use the word “terminal” throughout, and define a “terminal” as a device. In Book 4 Section 5.3 it is, however, stated that a “terminal” may be a distributed system, and in this case the word “terminal” applies to the whole system, however there are several other provisions which conflict with this statement¹. In other cases the specifications do not make it clear whether the word “terminal” applies to the EMV kernel, to the system as a whole, or to a specific component of the system.

Pragmatically, however, it is recognised by card schemes and acquirers that the hardware at the point of sale may comprise one or more physical components, and that other functions may be spread over multiple hardware units within the retailer network. Introducing more separate hardware units increases the complexity of testing and approval (see section 8 below).

The hardware at Point of Sale is likely to comprise one, two or three units relevant to the EMV application:

- a card reading /writing device
- a PINpad, and
- a point of sale terminal

These may be combined in several ways (for example, the card reader may be part of the PINpad or of the point of sale terminal); if they are all in one unit then this is outside the scope of this document.

All systems will also need a magnetic stripe card reader, not only to read non-chip cards but also for fallback purposes. This reader may be integrated with the chip card reader (either physically or controlled by a single device), or may form part of an existing device such as a keyboard. If the two are completely separate it may be less easy to manage the interaction between the two (see 7.1 c) below).

6.2 Card reader / writer

The card reading device (known in EMV as the IFD or Interface Device, or IFM – Interface Module) must be Type-Approved by EMVCo. This process is described in detail on the EMVCo website www.emvco.com. Vendors must register with EMVCo and pay a fee; they must submit their IFD to an approved laboratory for testing (also chargeable) against the current test specifications.

This “Level 1” testing process covers only the physical, electrical, timing and protocol aspects of the card reader. Manufacturers are strongly recommended to separate the IFD from other functions; for example:

¹ For example, the requirement in Book 4 section 10 that Terminal Type must be initialised at installation time.



Guideline G18 – Configuring Integrated Systems

- the IFD should be an identifiable separate hardware unit;
- any microprocessor or PAL functions within the IFD should include only electrical, timing and protocol-related functions;
- the IFD should be treated within the production system as a separate device (even when it is always sold as part of a PINpad or other device) and should have its own part code and version control;
- it should use components for which supply is regarded as highly stable, in order to minimise the risk of recertification (which might entail recertification of all kernels and systems using this device).

If changes are made to the IFD, recertification may be required (see section 8.8.1). EMVCo Type Approval Bulletin 11 [TA11] describes those conditions; in general, substitution of a passive component is regarded as a minor change, but changes to active components or to firmware are likely to be viewed as major changes and will require recertification. The vendor must describe the change to EMVCo, which will determine the tests to be performed.

6.3 PINpads

PINpads that do not incorporate a card reader/writer do not need to undergo EMV Level 1 testing. However, all PINpads (and devices incorporating PINpads) must undergo security testing; this is separate from the EMVCo tests, which test only functionality.

Up to 1st October 2004 manufacturers are required to comply with the Visa Offline PED Security Requirements (see www.visa.com/pin) and to be tested against the Derived Test Requirements (also to be found on that site). From that date the requirements are upgraded to the Payment Card Industry PED Security Requirements, which are also supported by MasterCard.

If a hardware, firmware or software change is made to a device that has a security certificate, then the change must be described to the laboratory that carried out the tests, and its impact evaluated. If it is considered that it could affect the security, further testing may be required.

PINpads for use in the UK must also be evaluated against the APACS PED Protection Profile by an approved security testing laboratory. The evaluation report must be submitted to the acquirer, who will determine whether the PINpad may be used in the environment proposed. Changes to the PINpad that affect its security-enforcing functions may require re-evaluation.

The nature of some of the security criteria requires that the PIN Entry Device be a physical as well as a logical unit. If the PED functions are contained within a larger device, that larger device must be tested. However, as with the card reader/writers, manufacturers will minimise re-testing by following these principles

- any microprocessor or PAL functions within the PED should include only functions necessary for security or for the correct functioning of the ISO 9564 PIN block encryption and decryption;
- the PED should be treated within the production system as a separate device and should have its own part code and version control – this applies whether or not it incorporates an IFD as described in section 6.2;
- it should use components for which supply is regarded as highly stable, in order to minimise the risk of recertification (which might entail recertification of all kernels and systems using this device).



Guideline G18 – Configuring Integrated Systems

6.4 Point of sale terminal

Where all the “EMV functions” (i.e. functions specified in the EMV specifications) are carried out in the terminal or PINpad, there are no further requirements imposed on the point of sale terminal. For example, a POS terminal that only sends an amount and transaction type (e.g. sale / refund) to the terminal does not have to be certified and can be freely changed without recertification.

However, where any EMV functions, including customer displays or cashier selections, are carried out in the POS terminal, then it is considered to be part of the “EMV terminal” and is subject to certification.

In nearly all cases, the testing is functional (see sections 7 and 8), however it should be noted that EMVCo level 2 type approval is given for a kernel running on a given software platform. EMVCo has clarified that this software platform is taken to be an operating system type (e.g. Windows 2000, Linux) and not a specific build, release or service pack of that software.

6.5 Other hardware components

Other system components, for example back-office systems that do not perform EMV functions, routers and other communications devices, are not specifically tested, nor do they require any certification or type-approval. However they will form part of the end-to-end testing described below in section 8.5).

6.6 Accessibility

All hardware designs for consumer-facing equipment should take into account, as far as is practical, the needs of disabled people – see Guideline G12 – Accessibility **[G12]**.

7 Software

7.1 Architecture

Good software design always involves a high degree of modularity. The BRC Chip Architecture **[BRC]** recommended a level of modularity based on the separation of device-dependent, application-dependent and business-dependent functions – see Figure 7.1. Software suppliers who have followed this architecture have found that it helps considerably when adding new hardware devices or applications. The architecture was intended to be completely general and would, for example, support Government cards or loading of transport cards as well as the payment cards (including electronic purses) and retail loyalty cards that were specifically envisaged at the time.

In Figure 7.1, the kernel is taken to include only the card and application interfaces, and not the business functions. For a more detailed description of the functions included in these Services, see Appendix A.

Guideline G18 – Configuring Integrated Systems

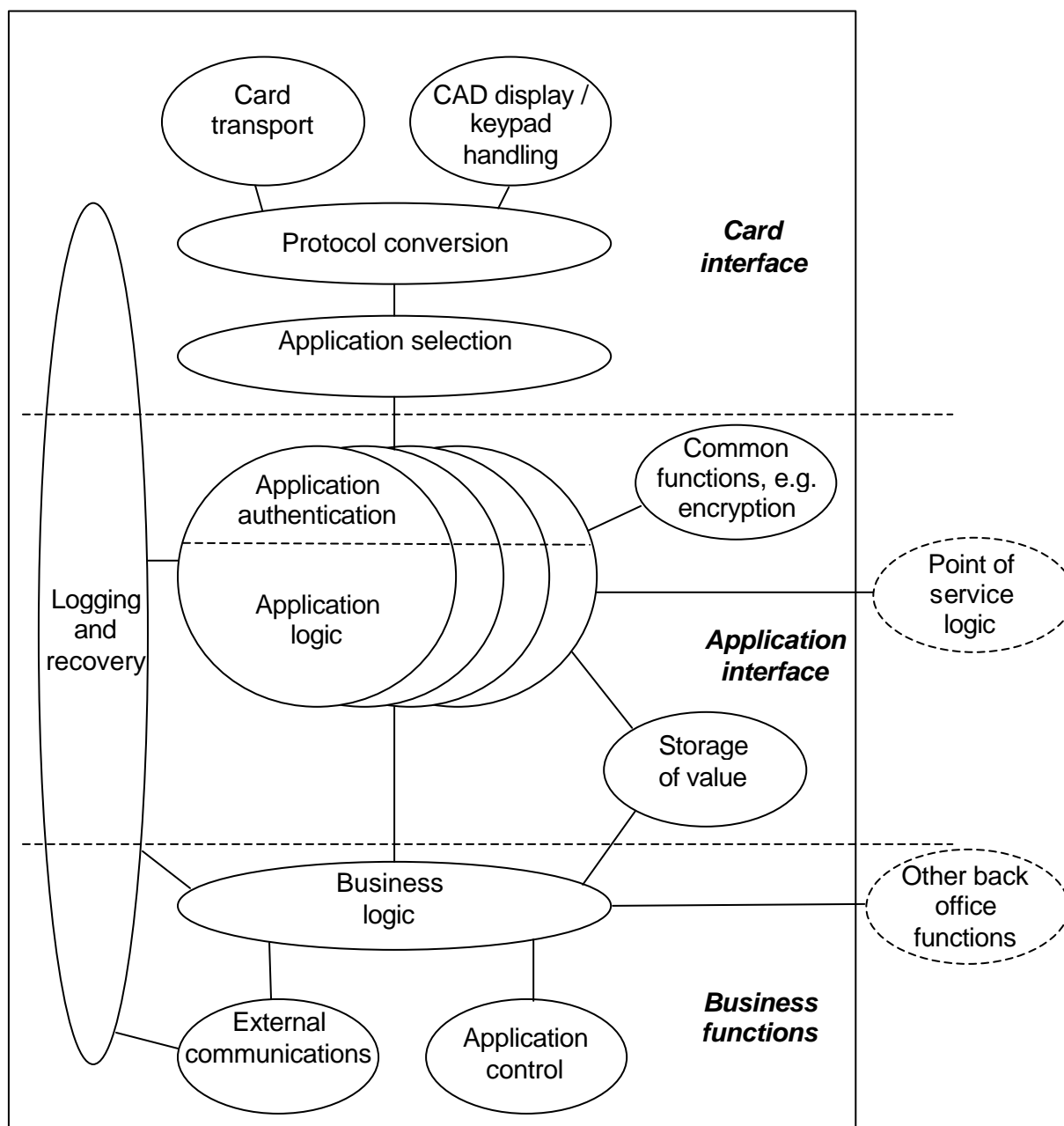


Figure 7.1 – BRC chip architecture functional components

Experience has shown that this general architecture remains valid. However, in the specific case of EMV, experience has led to some qualifications that should be understood by developers:

- a) the interface between the Application Logic and the Business Logic has been shown to be particularly sensitive; for example, when an application is Unable to Go Online, the procedure to be followed must follow Card Scheme product rules and any agreements made with the acquirer, as well as the retailer's own business needs. The Business Logic (which is assumed to be outside the EMV kernel) must follow these rules and will be subject to scheme and acquirer testing during the integration phase.
- b) ISO 9564 encryption and key block construction are considered by EMVCo to be EMV Level 2 functions; where these are performed by the PINpad, the PINpad



Guideline G18 – Configuring Integrated Systems

firmware forms part of the EMV Kernel and any change of or to the PINpad requires recertification at Level 2.

- c) Many vendors have experienced difficulty in integrating magstripe functionality into this architecture. This is not because of any difficulty in handling magstripe cards, or of integrating them into the business logic, but because the rules of the international card schemes impose very specific conditions on the handling of fallback transactions. A magnetic stripe transaction may only be initiated at a chip-capable terminal if:

- The magstripe does not carry a service code indicating the presence of a chip, or
- A chip read has been attempted and failed

It is the view of the authors of this document that these relatively simple rules should be built into the magstripe application (which should then be kept separate from the chip application). A flag or indicator is required to show that a chip read has been attempted (as indicated by a hardware switch in the card reader) and failed (no ATR or unable to select an application). If the card reader hardware itself fails, the reader should be disabled (and the action logged).

The system adopted in France for fallback transactions can then be used: if a chip read fails, the terminal displays “Card read failure” and the operator has 30 seconds to initiate a magstripe read.

- d) The BRC document further recommended a number of possible Open Standards for interfacing between the modules; these have not been widely adopted and indeed where manufacturers have published APIs, EMVCo has not been prepared to recognise these in determining the tests required. It therefore appears that using an open interface standard will not reduce certification costs; however it should reduce support costs, since there will be fewer software versions to support.
- e) It is critical both for support and certification that the boundary between kernel and non-kernel functions is drawn correctly. If the number of functions in the kernel is too great, then too many versions will require certification, and small changes in business practice may require recertification. From the point of view of the integrated systems supplier, only the minimum number of functions should be contained within the kernel, and a large number of “hooks” should be provided to allow non-kernel functions to control the transaction process flow between each stage of the EMV process.

7.2 Parameters

It is also a tenet of good software design that code should be code; any parameters should be held outside the code. Changes to the behaviour of the software should be effected by changes to these parameters, and software to change the parameters should be defined and tested like any other software.

At present EMVCo does not recognize any difference of principle between code and parameters. Changes to **EMV parameters** (parameters that are defined by EMV and, in most cases, have a tag value or are contained within an EMV-tagged data element) are regarded as changes in the software. EMVCo has indicated (in General Bulletin 19) that it will introduce a form of testing that will allow multiple parameter sets to be tested in a single set of laboratory tests, however details of this are still not available at the time of writing.



Guideline G18 – Configuring Integrated Systems

Vendors have much more freedom with parameters that are not specifically defined by EMV (i.e. that do not have an EMV tag and do not form part of a data element that has a tag). This could, for example, cover alternative ways of handling PIN bypass transactions (allow, call supervisor, do not allow). Changes to these parameters do not in principle require recertification, although where they affect the behaviour of the business logic they should be agreed with the acquirer.

The following sections give further guidance with regard to specific parameter groups.

7.2.1 Floor limits and associated parameters

Because of differences in product rules between schemes, and sometimes between products within a scheme, it is strongly recommended that the following parameters be held at AID level (with a further set of parameters for magstripe transactions):

- Acquirer identifier (and where necessary associated communications parameters)
- CA Public Key
- Merchant Category Code (see [APACS70] for rules re cash disbursement)
- Target, threshold and maximum percentages for Random and Biased Random Selection
- Terminal Action Codes
- Terminal Floor Limit

7.2.2 Terminal Type

Most terminals of the type considered here will in the UK be set up as terminal type “22”: Merchant-controlled, offline with online capability. Some terminals may also be set up as Type “25” (Unattended merchant-controlled with online capability).

There is a potential issue where parts of the system (and hence the “terminal”) are shared between points of service with different capabilities or of different types. For example, a common Terminal Risk Management service might control some attended and some unattended points of service. In these cases the system must be so designed that transactions from the attended points of service always use one set of parameters, and those from the unattended points use a different set. How this is achieved is a function of the system architecture; it would be feasible to have a simple mapping table of points of service to terminal type. This should be transparent to the integration testing process and should not affect EMVCo certification.

7.2.3 Terminal Capabilities

It is an EMV requirement that Terminal Capabilities and Additional Terminal Capabilities be initialised in the terminal before the start of the transaction. This is normally interpreted to mean that they may not be varied based on a transaction parameter (such as the card AID), and it is therefore assumed that there must be one set of Terminal Capabilities and Additional Terminal Capabilities parameters for the terminal.

As discussed in the previous section, any transaction from a single point of service should always use the terminal capabilities appropriate to that point of service.

Byte 1: Card Data Input

In the UK, terminals should normally be set up to permit Card Data Input from Magnetic Stripe (bit 7) and IC with Contacts (bit 6). For most *attended* terminals, Manual Key



Guideline G18 – Configuring Integrated Systems

Entry (bit 8) should also be permitted (it may be disallowed through business logic and prompt sequences for most transactions if required).

Recommended Value: E0

Byte 2: CVM Capability

UK Chip and PIN attended terminals must support Plaintext PIN (bit 8), signature (bit 6) and enciphered offline PIN (bit 5). Some acquirers may, exceptionally, permit enciphered online PIN (bit 7). Unattended terminals should support “No CVM” in order to meet Visa requirements.

Recommended values: Attended B0, Unattended B8

Byte 3: Security Capability

All UK terminals *must* support SDA (bit 8) and DDA (bit 7). They *may* also support CDA (bit 4). Motorised terminals *may* support Card Capture (bit 6) but this is not a requirement.

Recommended value: C0

7.2.4 Additional Terminal Capabilities

Byte 1: Transaction Type

Most merchant-controlled terminals should support Goods and Services only (bits 7 and 6); if cashback is or may be offered, then bit 5 should also be set. Note that cashback is not permitted on some products; this is determined using the EMV logic. The retailer may, however, control - using its business logic, the prompt sequence and cashier training – whether the customer is offered cashback proactively and whether it is permitted at all transaction values.

Recommended values: 60 (or 70 if cashback is supported)

Byte 2: Transaction Type Cash Deposit

This is not relevant to merchant-controlled POS terminals and should be set as 00.

Byte 3: Terminal data input capability

All terminals should have numeric keys (bit 8) and function keys (bit 5)². They may additionally have alpha and/or command keys (bits 7 and 6)³. These parameters could in principle be passed to the card but are not used in EMV processing.

Recommended value: 90

Byte 4: Data output capability

Most terminals should have a capability for attendant and cardholder printing, attendant and cardholder display (bits 8, 7, 6 and 5). See next paragraph for code table bits.

Recommended value: F0

Byte 5: Data output capability (code tables)

² It is assumed that the CANCEL, CLEAR and ENTER keys are considered to be function keys; there may also be other function keys.

³ The numeric, cancel, clear and enter keys must conform to one of the two layouts specified in ISO 9564; other keys should be physically separated from this main keyblock.



Guideline G18 – Configuring Integrated Systems

Most English language-only terminals need only support the Common Character set and therefore need support no code table. Multilingual terminals will often support Code Table 1 (ISO 8859-1 or Latin-1) or other code tables. It is advantageous to obtain approval with at least one code table (unless storage space is very limited) as this allows the code table to be changed later. If no code table is supported, then a major change is required to support one.

The minimum recommended value for this byte is therefore 01, however 00 is permissible and this will result in the smallest number of tests; further code tables may be supported.

7.2.5 Other ICS options

The following options may be stored as parameters or may simply be functions hard-coded into the kernel:

Option	Recommendation for UK terminals
Support PSE selection method	Recommended
Support Cardholder Confirmation	Recommended
Does terminal have a preferred order of displaying applications?	Terminals may list applications in order of frequency of use, alphabetically or by other criteria. The former is preferred.
Partial AID support	Mandatory
Multi language support	Optional – depends on environment and location
Common Character Set support	Mandatory
Maximum CA public key size	Minimum 2048 bits mandatory
Supported exponents	3 and $2^{16} + 1$ mandatory
Check revoked Issuer PK certificate?	Recommended
Default DDOL	Mandatory
PIN bypass support	Recommended for attended terminals
Support Get PIN Try Counter	Recommended
Support Fail CVM	Recommended
Support No CVM	Mandatory for unattended terminals; not recommended for attended
Support Amount Other Field	Recommended; mandatory if cashback supported
Floor limit checking	Recommended (mandatory unless online-only)
Random Selection	Mandatory unless online-only or offline-only
Velocity checking	Recommended



Guideline G18 – Configuring Integrated Systems

Transaction Log	Recommended (size dependent on function)
Exception file	Recommended (dependent on function)
Support TACs	Recommended unless online-only
Offline-only terminals: Process Default IAC/TACs before/after 1 st Generate	Recommended for offline-only terminals
Forced Online Capability	Recommended for attended points of service
Forced Acceptance	Not recommended
Advice support	Dependent on capability – required for APACS 40 and equivalents
Referral support	Recommended for attended terminals
Batch data capture support	Mandatory unless online-only
Support Default TDOL	Recommended
Exception handling	Magstripe mandatory for attended

7.2.6 Unable to Go Online

There are several parameters which, taken together, determine the retail system's response when Unable to Go Online. The most important parameters are:

- Terminal Action Codes
- Floor Limits
- Random Authorisation levels

Terminal Action Codes for each AID are generally mandated by the relevant card scheme. However, retailers that are prepared to assume some risk during a comms failure may apply for a waiver from their acquirer; this applies particularly to the TAC for "Over floor limit – default" where it has been common practice in the UK to apply a higher "post-comms" floor limit in this situation. Retailers who currently apply a post-comms floor limit are advised to discuss this with their acquirer.

It was recommended above that floor limits be held at AID level; this allows, for example, different levels to be set for debit and credit transactions. Merchants who have operated zero or very low floor limits in a magnetic stripe environment should consider reverting (with the agreement of their acquirer) to the sector floor limit, and using a high Random Authorisation Level to achieve the level of authorisations sought. This makes it clear to both issuer and card that the authorisation is sought voluntarily by the retailer, and is less likely to lead to a decline. In addition, a transaction that has been sent online for Random Authorisation will almost always be accepted by the card if the terminal is Unable to Go Online, whereas an Over Floor Limit transaction is much more likely to be declined or to call for a voice authorisation.



8 Testing requirements

8.1 Pre-Certification Testing

Before submitting system components for certification, at whatever level, the developer must have very high confidence that it will pass at the first attempt. Type approval in the lab or with the acquirers must not be seen as a debug opportunity. Retests are both time-consuming and costly. Success can only be achieved through thorough preparation, understanding of requirements and testing.

Readers should also refer to Programme Guideline G6 – Certification Process Description.

It is expected that retailers/system integrators will select card acceptance devices that have already passed EMV Level 1 certification. However, for completeness, pre certification testing for Level 1 is included in this document.

Test tools are available that perform the following tasks:

- Card simulation
- Electrical and protocol tests for EMV Level 1
- Interception and interpretation of dialogue between card and CAD
- Acquirer host simulation
- Issuer host simulation

8.2 EMV Level 1

EMV Level 1 certification focuses on the card acceptance device and the high-level objective is, to ensure that the terminal does not damage the card and that it complies with the specifications defined in EMV Book 1 Part 1.

EMVCo Type Approval documentation includes a set of test cases that must be successfully performed. The EMVCo Test case document details the tests and the standard for equipment needed to perform them.

Tools are available to analyse the communication signals and protocols (functional) and electrical.

8.3 EMV Level 2

EMV Level 2 certification focuses on the software application (EMV Kernel) to ensure that it conforms with EMV specifications. Test cases are published by EMVCo that must be successfully executed. Two sets of test cases are required from EMVCo – EMV version 4.0 Test Cases and EMV Test Case Errata to EMV 4.0. Both sets must be used.

Where any part of the level 2 functions is carried out within the PINpad, the level 2 type approval must include the PINpad; where the PINpad only performs PIN entry, display and local encryption any approved PINpad (including level 1 approval where appropriate) may be used (see EMVCo Type Approval Bulletin no 11).

To perform the test cases, test tools are required that will emulate the test cards; these can be through software emulation or by download to programmable test cards. There are tool providers that offer either solution – see section 8.6 below.



Guideline G18 – Configuring Integrated Systems

The EMV accredited test labs use tools, accredited by the card schemes, to perform the accreditation tests. Vendors are recommended to contact the labs to determine what tools are used.

When testing, the test tool setup should be designed to isolate the EMV Kernel, and some additional test functions may be needed in order to exercise the kernel completely. Input and output channels to the kernel should be analysed; this demands low-level debugging tools and log output.

8.4 Application Functional Testing

Application functional testing is the responsibility of the application integrator. It is anticipated that it should be carried out in a test environment. The test specification may be written by the card scheme or its agent, or by the application integrator. Test cases provided by the card schemes (MasterCard ETEC and Visa ADVTK) should also be used. These may be provided either as card sets, purchased from the schemes or as part of a test tool kit (card simulation). Acquirer and issuer host simulation will also be necessary.

The application functional tests are not performance tests, and the fact that simulators and hardware other than the target hardware are used mean that any performance results must be interpreted with care. Specific tests may and should be performed to assess the likely performance of the target system, but achieving the required performance is the responsibility of the application integrator and can only be demonstrated at the integration stage. Depending on the network and software configuration, there may at this stage be a requirement to simulate loading of the application software and related systems.

The POS process steps should be carefully tested in conjunction with card scheme rules and **[APACS70]**. Experiences to date have identified Referrals and Refunds to be processes where problems have been encountered.

8.5 Bank Accreditation

A retailer will need to have its system accredited by each acquirer they use. In most cases this will be just a single entity but other retailers may split their acquiring across schemes or brands, and if they accept American Express, JCB or (in the future) Diners, these schemes will also carry out accreditation tests.

Testing should be carried out to satisfy the requirements of the acquirers. Scheme specific certification (Terminal Integration Processing for MasterCard, Interoperability Testing for Visa) will also need to be carried out during this phase.

Early contact with the acquirers is recommended, to ensure that the test requirements are known. Acquirers may be able to make available specific test cards for testing with their acquirer hosts. Retailers should ensure that there is one person with responsibility to co-ordinate and liaise with all relevant parties, including the acquirer(s), schemes, suppliers and installers. This person must have sufficient control and influence over the other parties to ensure that timescales are met and that responsibility for fixing problems is correctly assigned and accepted.

Common errors encountered include:

Guideline G18 – Configuring Integrated Systems

- ❑ Ensure PED and POS messages make sense in the context (e.g. 'Card read failure' where appropriate instead of 'Card Reader Failure')
- ❑ Incorrect flagging of fallback
- ❑ Not checking results properly
- ❑ Script truncation in authorisation response – should not normally occur
- ❑ The communications protocol (e.g. ISDN, X25) used for acquirer acceptance testing must be the same as that to be used in the live environment

8.6 Test Tools

Figure 1 provides a list of test tool vendors. However, the Chip and PIN Programme is vendor neutral and makes no recommendations of any product. The list of vendors has been compiled based upon knowledge that the Chip and PIN Programme has gained; neither the PMO nor Teconomica Ltd makes any warranty as to the completeness or accuracy of this list.

Test Tool Provider	Contact Details	Card Personalization	Card Simulation	Terminal Simulation	EMV Level 1	EMV Level 2	Card/ device dialogue capture / analysis	Acquirer simulation Domestic protocols	Issuer simulation
Aconite	www.aconite.net 0870 241 5841		Y	Y		Y		Y	Y
Acquirer Systems	www.acquirer.com 07971 899783	Y		Y		Y	Y	Y	Y
Card-Right	www.card-right.dk +45 45 88 92 32				Y				
Collis	www.collis.nl +31 71 581 36 36			Y	Y	Y	Y		Y
ICC Solutions	www.iccsolutions.com 01925 629001		Y	Y		Y	Y		Y
Integri	www.integri.be +32 2 717 69 50	Y	Y	Y	Y	Y	Y		Y
KaSys	www.kaSys.net +33 1 46 99 69 00		Y	Y		Y	Y		
MasterCard EVAL	vendors@mastercard.com					Y	Y		

Figure 2 - Test Tool Providers

8.7 Common Problems Encountered

Labs and acquirers have encountered a number of problems with systems submitted for test. Many of these could have been prevented by:

- ❑ Proper preparation
- ❑ Better planning
- ❑ Early engagement with the labs and acquirers
- ❑ Understanding the implications of configuration changes
- ❑ Full understanding of the specifications, requirements, test conditions, operational processes
- ❑ Use of appropriate test tools, understanding their use and interpretation of results



Guideline G18 – Configuring Integrated Systems

8.8 Recertification

In general, once authority has been given to a product or retailer system, it may be used without limit. Recertification is, however, required under various conditions as detailed below:

8.8.1 EMV Level 1

Recertification is required if any layout or active components are changed. EMV states (in [TA11]) that “most changes to an IFM are classed as major changes”. However, it is the supplier’s responsibility initially to assess the change and notify EMVCo accordingly. Suppliers may request an opinion from EMVCo, however this almost always results in a view that the change is major.

Major changes require recertification against the current criteria and test scripts; this means that if a vendor has previously obtained type approval against an earlier version of the specifications that is no longer available for new approvals, a new approval must be obtained against the current criteria. If any test scripts have been added or modified in between, the new scripts will be used.

8.8.2 EMV Level 2

The Appendix to [TA11] shows which parameters within an ICS are considered to constitute major changes. The kernel requires recertification if any of the following parameters are changed or added:

- CVM Capability
- Security Capability (excluding Card Capture)⁴
- Transaction types Cash, goods/services, cashback
- Code table support on/off (but not change of a table)
- Application selection (but not language support)
- Data authentication
- CVM support
- Terminal risk management
- Terminal Action Analysis (but not values in the TAC table)
- Completion processing

In addition, a change of operating system (but not an update or new build) or a recompilation of the kernel requires recertification. The inclusion of “recompilation” in this list has caused significant dissent⁵ and this is being discussed at meetings between vendors and EMVCo.

As mentioned in section 6.3, PINpads that perform ISO 9564 encryption and PIN block construction (i.e. virtually all separate PINpads) are considered to be performing a level 2 function; a change of PINpad therefore requires recertification. EMVCo is currently discussing a formula that would allow a supplementary set of tests (rather than a full recertification) to be carried out when a component such as this is substituted.

⁴ It is not clear whether removing CDA support would be a major change, but adding it presumably would be

⁵ Under many systems, correcting a spelling mistake in a prompt could require recompilation, although this is not a “significant change in functionality” as specified in [TA]



Guideline G18 – Configuring Integrated Systems

8.8.3 PINpad security approval

Changes to PINpads are subject to review by the laboratory that carried out the initial assessment. It may agree that the change does not affect any tested functions, ask for supplementary tests or a full re-test.

8.8.4 Bank accreditation

Each retailer system differs slightly from that of every other retailer. An EMV transaction passes through, and is potentially affected by, many separate systems in each retailer and there is therefore scope for wide differences in behaviour.

In practice, most aspects of technical interoperability are attributable to the combination of type-approved components (IFD, kernel and PINpad). Some further aspects, mostly relating to the cardholder and cashier experience and to product rules, are determined by the prompt sequences and business logic in the point of sale system. Terminal risk management may include functions such as hotlist checking carried out in a head office host system.

From the point of view of MasterCard and its integration testing, there is a critical “similarity test” under which a system may undergo a much reduced set of tests provided it uses the same type-approved components as another system that has already been approved.

Acquirers will take a more holistic view of the system and will determine whether the key components, in particular those that determine the transaction flow, have changed. Running end-to-end tests (using both card scheme and their own test packs) is a key part of this testing and every system will always undergo some testing.

Acquirers will take into account testing completed with another acquirer, but there are differences between acquirers, particularly in submission formats and communications options; these areas will therefore be tested more thoroughly by each acquirer.

Once a system has completed bank accreditation, recertification is not normally necessary unless a significant change is made to the system. A “significant change” is a software change that affects the kernel or logic of transaction handling, or a change of hardware or operating system in the card reader/writer, PINpad or terminal on which the EMV kernel runs. If this happens, acquirers will agree a set of tests to be carried out.

However, scheme Letters of Approval and PINpad security certificates do have expiry dates; the purpose of these dates is to allow review of the system if significant external changes have taken place (for example, a new cryptographic technique might allow keys stored in particular ways to be broken). It is not expected that retailers will be refused approval to continue accepting transactions at these expiry dates, however vendors should liaise with schemes and labs to minimise the risk of problems at expiry.

9 Conclusions

Retailers in the UK and many other countries have developed highly distributed and configurable systems for managing their business, including acceptance of card payments. It is possible to retain this architectural principle when implementing EMV, although there are some limitations as to what components can reasonably be placed where.



Guideline G18 – Configuring Integrated Systems

Adopting a structure similar to that proposed in the “BRC Chip Architecture” of 1999 will help to minimise the effects of the complex structure of type approvals and certifications that have proved necessary.

Hardware designers should select designs and components with a view not only to price and function, but also take into account longevity and flexibility.

On the software side, particular attention must be given to some interfaces and interplay between functions, such as the implementation of fallback to magnetic stripe and the process when Unable to Go Online.

The storage of parameters and choice of optimum parameter sets also requires special care. This paper includes guidance on the parameters that are recommended for most UK implementations. It is often better to support a function in the EMV kernel, and then not call it from the point of sale software, than not to support it in the kernel. This is because any subsequent change in functionality, or using the same kernel in another retailer, requires less testing.

For each retail system, one organisation – and preferably one person – must take responsibility for the architectural design and the integration of components from different suppliers; they must also co-ordinate the bank accreditation phase.

The first implementation of any one system is complex and a large amount of testing is required to ensure widespread technical interoperability and compliance with scheme and product rules. However, if a system has been carefully designed with multiple implementations in mind, it should be possible to reduce the amount of testing substantially for subsequent implementations.



APPENDIX A

Functional Services described in BRC Chip Architecture

(updated June 2004)

Card transport

The function of this service is to ensure, as far as is possible within the constraints of the operating environment, that the card is positioned and provided with power and clock signals as directed by the Protocol Conversion service, and to communicate the position and electrical status of the card to the Protocol Conversion service.

For EMV, this forms part of the EMV Book 1 / Level 1 specification and is not repeated here.

Implementors should also consider the potential need for contactless and motorised readers. Motorised readers for use in vending should have at least two status signals for levels of card insertion (e.g. 25%, 95% and 100%) on the input side and one on the output, as well as inputs for motor forwards/reverse. Other signals (e.g. shutter open/close) may also be provided. In the case of contactless readers, the card transport service includes the provision of RF power and "card present" detection.

It is permissible for Card Accepting Devices to have two modes, starting up in the mode described in ISO 7816-3 (1997) or in ISO 7816-10 when a non-payment card is expected, changing to the "EMV" mode when a payment card is expected. This must be transparent to any EMV card operation.

Display and keypad handling

The Display and Keypad Handling Service displays messages to the cardholder and accepts keypad input from the cardholder.

For EMV, the functions are specified in EMV Books 2, 3 and 4. PINpads are also⁶ subject to security requirements as specified in the Payment Card Industry PED Security Requirements **[PCI]**.

There are some issues where a customer display and keypad are required other than for PIN entry. In these cases two instances of the Service may be required (e.g. in outdoor petrol terminals it may be necessary to have a separate keypad and display for mileage and registration number entry). This is because it is usually easier and cheaper to provide two keypads than to meet the security requirements for a combined unit.

Protocol conversion

This Service communicates with the card. For EMV it forms, along with the Card Transport Service, the Book 1 / Level 1 component.

Where memory cards may be used, the service should also be capable of handling memory cards meeting the ISO 7816-10:2000 standard.

⁶ From 1st October 2004 for new PINpads; up to that date the Visa-only requirements apply



Guideline G18 – Configuring Integrated Systems

Where the Card Transport also handles magnetic stripe cards, there may be an equivalent service alongside the Protocol Conversion service for handling the magstripe data. This service will decode analogue F/2F data from the relevant tracks, buffer the data and convert it to 8-bit characters in either TTL or "RS232" serial form. This data is normally passed directly to the application; it may use either a separate interface from the chip data or the same interface. In the latter case, the Application Selection service must pass the data through to the application transparently. This permits magstripe and chip cards to be handled within one homogeneous architecture. These aspects are not considered further in this document⁷.

Application selection

The function of this service is to select a Card Application (for cards meeting the EMV Card part II specification or ISO 7816-4 and ISO 7816-5) or to pass control to a special application for memory cards.

A recommended sequence of selection is as follows:

- The Point of Service application determines, from the context of the physical transaction, which card application types would be relevant at the current point in the transaction. When only one application is appropriate, it may select (call) that Application Logic directly; in other cases it must select the Default Application Logic (see section 0) and pass the list of applications or application types via that service to the Application Selection service;
- The Application Selection service determines which applications (within these types) are supported by the retail system;
- The Application Selection service determines which of these applications are offered by the card (by the use of the EMV Payment Systems Environment and ISO 7816-5 commands)
- For EMV cards, the rules for Application Selection are contained in Book1, Section 12 of **[EMV]**. However, terminals must also follow the requirements of Section 6 of MasterCard's M/Chip Functional Architecture **[MCFA]** and of section 3.5 of **[VIS]**. These provide for elimination of an international application from the candidate list where a competing domestic application is found.

If, during ATR processing, a memory card is detected, the service shall pass control back to the relevant Application Logic service, giving as data the content of the ATR.

During the transaction, the Application Selection service is transparent to the communication between the Application Logic and Protocol Conversion services.

When the card is closed (either by being withdrawn or explicitly turned off) all applications are deselected and the application list should be cleared. (Applications may continue to run until they terminate, particularly if there are exception conditions).

Application Logic

a. Default Application Logic

Standard operating procedures may dictate that the card application, and therefore the corresponding Application Logic service, be manually selected. In such cases, a default Application Logic service must be in place to adopt the focus of control and manage the process

⁷ But see comments in section 7.1

Guideline G18 – Configuring Integrated Systems

of application selection. Many of the general requirements of a standard Application Logic service apply equally to the default, but there are some additional requirements for this service.

As part of the general error and exception handling, the default Application Logic may be faced with a situation where a card authentication has failed and no particular Application Logic service could be invoked. In this case the default Application Logic must call upon the facilities of the surrounding services to inform the cardholder and/or cashier of the problem and take appropriate recovery actions.

A more detailed view of the interface between Application Logic and Application Selection is shown in Figure 2.

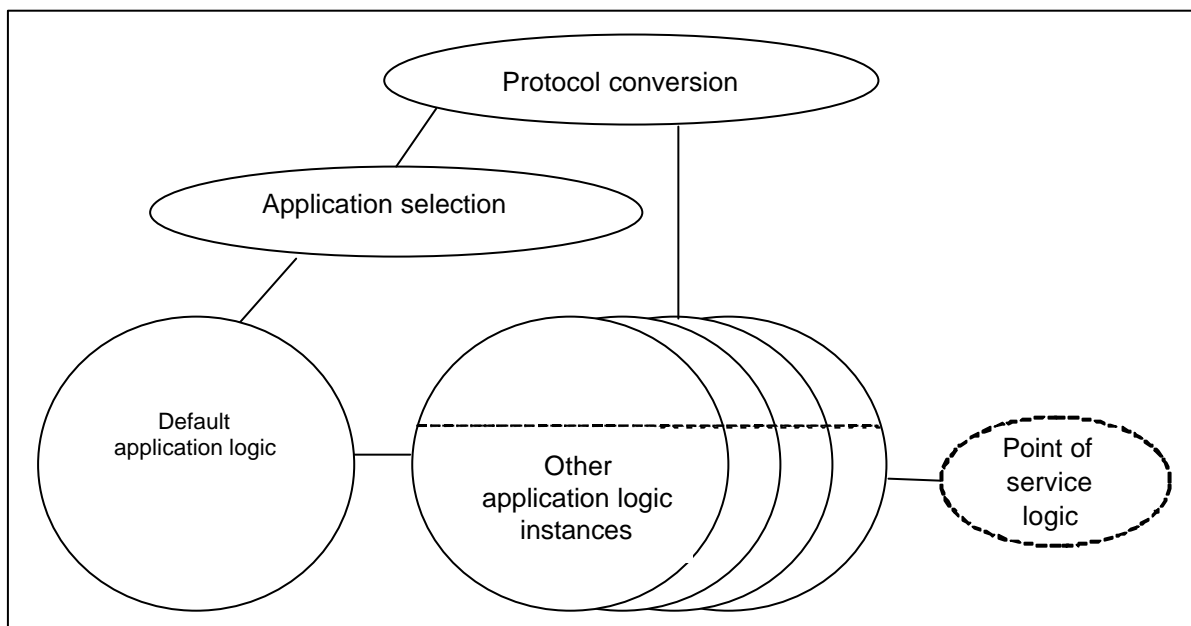


Figure 2: Default Application Logic interfacing

b. Functions

The Application Logic service provides all the functionality necessary to support a specific application within the card. It adopts the primary focus of control and collaborates with the surrounding services to provide this functionality.

There is one instance of Application Logic for each card application supported in the system, and a Default Application Logic which is operational when no card application is selected. An Application Logic service may also be an applet downloaded on a periodic or on-demand basis.

After Application Selection has been performed, Application Logic communicates with the application on the card and has access to the CAD Display/Keypad. It is also required to react appropriately to exception conditions, such as premature card removal.

Application Logic has access to the generic set of shared library functions within the Common Functions service but it must supply all the data relevant to these operations because the latter do not have any data storage facilities.

Application Logic services may be chained or called from within another Application Logic service.

Application Logic will inform the POS Logic service:



Guideline G18 – Configuring Integrated Systems

- when the card is in place or removed
- when there is a list of applications from which a manual selection must be made
- when it needs to know the transaction value, or fraction of the transaction value in the case of a split tender
- when it has a transaction related result, e.g. authorisation response
- when it wants to write to the POS display
- when it wants to read the POS keypad
- when it wants access to any other of the POS peripherals
- when it wants a decision on a cancellation or reversal

Application Logic also controls the interface between the Application Logic and the Storage of Value service; Application Logic will invoke the Storage of Value service when value is to be stored or redeemed in accordance with the rules of the relevant scheme.

Application Logic will inform the Business Logic service:

- when it seeks an on-line authorisation
- when it wants to log a transaction
- when it gives permission for product to be dispensed
- when it wants access to sensitive data (but this must be carefully controlled)

This service shall co-ordinate (and where necessary initiate) the recovery of other services following an exception condition. Some applications have specific and proprietary error recovery rules (but they must be checked for completeness in a distributed environment).

c. Specific requirements for the Default Application Logic

The Default Application Logic must manage the process of application selection when:

- Card authentication has failed or no application could be selected
- Standard operating procedure dictates a manual selection
- An applet needs to be downloaded to process the card's application
- There is no Application Logic service in place to process a card application

The Application Control service knows about all the Application Logic services which exist, their version number, their associated authentication keys and whether they are an applet which requires a download. Where necessary, the default Application Logic must obtain the relevant information from Application Control.

d. Card Application Authentication

Authentication of the application on the card (often called Card Authentication) is frequently (with single function cards almost invariably) an integral part of the application.

There is a practical advantage in a checkout environment in performing card or application authentication as soon as possible, since this part of the operation can be removed from the critical timing path. This would require a change from current procedures for bank cards, but is strongly recommended for all cases where it is permitted. Card authentication may be carried



Guideline G18 – Configuring Integrated Systems

out in parallel with other POS activities and with other Application Logic functions, and may therefore be considered by some applications as a separate service within the Application Logic service. Card authentication is valid only for as long as the card remains in the reader.

e. Application Logic services for memory cards

The requirements for Application Logic services for memory cards are identical to those for microprocessor cards. They may interface with the card through a driver (preferably PC/SC or OCF), and this is the recommended form of operation. In this case the PC/SC or OCF driver is seen as an additional Application Logic instance.

Common Functions

The Common Functions service provides a library of common functions which can be shared by all applications.

Such functions are likely to include MD1 Message Digests, DES encrypt / decrypt (single and triple), RSA signature / verify. The minimum set is that required by Annexes E and F of the EMV Card Specifications. A further recommended list of common encryption functions is given in PC/SC part 8.

In common with each Application Logic service, the Common Functions must manage sensitive data in a satisfactory manner at all times and all the same rules apply: e.g. keys and buffers should be explicitly emptied after use to ensure no cryptographic residue remains which could be used fraudulently. However, unlike the Application Logic services, there is no persistent storage of data; all such data storage requirements must remain within the appropriate Application Logic service.

This service has a responsibility for the specialist interfacing and management of any hardware associated with its operations, notable cryptographic co-processors and the like. The latter should be abstracted from the interface as seen by the Application Logic services.

Point Of Service Logic

The Point of Service (POS) Logic is the existing software which controls the interface with the Point of Service peripherals (keyboard, display, weighing machine, scanner etc) and with the other retail functions. It controls the flow of the sale or service transaction, and must be aware of the functions of the card system (but not necessarily of the details of any one Application Logic service). As such it falls outside the scope of the chip card acceptance implementation, but the interface to this service is one of the most important.

POS Logic must work to the API defined by the Application Logic service(s). There may be many such services with different APIs; it would be desirable for all payment applications, for example, to work to a common API, but no such definition exists today. The JavaPOS⁸ standard includes an API for a range of devices, including ISO 9564-compliant PINpads, but does not include any APIs towards upstream (business-focused) processes.

The Point of Service logic will initiate (call) the Application Logic or Default Application Logic service, passing any parameters such as an amount. Thereafter the Application Logic runs independently, calling on POS Logic for keyboard input, operator display output or other data as required. It returns control to the POS Logic when it terminates (successfully or otherwise). The only valid intervention by POS Logic during this process is to cancel the card transaction.

⁸ See www.javapos.com



Guideline G18 – Configuring Integrated Systems

POS Logic should therefore be able to accept and handle unsolicited messages for display and requests for information from the keyboard, although if such use of these peripherals is inappropriate at the time, suitable exception responses should be given. The same applies for other peripherals that may form part of the POS. When the POS wants to communicate unsolicited instructions to the Application Logic, e.g. a cancel request, this should be done via the support of the underlying operating system or simply waiting until the Application Logic next polls for information.

Initially the default Application Logic service will fulfil the link between the POS Logic and the rest of the system (see Figure 2). It will actively involve the POS in the manual selection of the application and handling of card authentication errors.

The POS Logic performs a vital rôle in connecting the standard POS transaction processing equipment to the new technology facilities of the chip card architecture. This fundamentally important responsibility must embrace the handling of error and exception conditions which will inevitably arise from user and application errors.

Storage Of Value

This Service is not required for EMV payment cards and is not considered further here.

Business Logic

Business Logic forms the interface between the card-related functions and the "back office" functions of the retail system (i.e. all functions other than the POS Logic).

It has a wide range of responsibilities and will itself comprise many sub-modules and components. In any one existing retail environment, many of these components will probably already exist in some form for handling existing payment and loyalty cards.

The functions of Business Logic include:

- The commercial processing of any payment transaction and its routing to a host system or acquirer:
- Posting of the transaction to the retailer's own system
- Secure transaction storage (from the time of the transaction until confirmation is received from a host system that they have been entered and processed)
- Subsequent processing of the transaction within the retailer's system (e.g. customer or transaction data analysis)
- Risk management, other than those functions specified by EMV
- Generation of on-line authorisation request parameters. It is recommended that a common bit-mapped interface (e.g. *prENV1750* or its UK implementation *APACS 60*) be used at this level and that conversion to other standards be considered the responsibility of the Host Communications service.
- Selection of authorisation host (where applicable)
- Interpretation of on-line response
- Handling of reversals (for completed transactions)
- Terminal error handling



Guideline G18 – Configuring Integrated Systems

- Network error handling
- System error handling
- System initialisation: this may be a substantial task, including setting up lists of supported Application IDs and version numbers (with reference to Application Control);
- Handling requests to download applets
- End of day functions, including:
 - file transmission, reconciliation and parameter downloading for credit & debit card systems;
 - unloading and reconciling the secure value store for electronic purse systems.

Logging And Recovery

The Logging and Recovery service is a centralised facility for logging all errors and exceptions. It may take the form of a single file, a set of files or a relational database. Each record will be tagged with a unique transaction reference ID so that all records relating to any particular transaction may be retrieved on request. Although it is termed the Logging and Recovery service, its rôle within the latter is predominantly passive: other services will interrogate the log and take the corrective action. It is primarily the rôle of the Application Logic service to co-ordinate recovery.

If the underlying operating systems allow, Logging and Recovery should signal to the active Application Logic service whenever any other service has requested that a record be logged. This is to facilitate recovery, which is co-ordinated by Application Logic.

The Logging and Recovery service may also be used for reporting and statistics purposes but any such use is outside the scope of this document.

External Communications

This service is responsible for high-level communications, including the interface to an acquirer or other external system. It will use existing communications paths and may incorporate many common functions with other applications. It should include:

- Conversion of on-line message formats (from the common standard referred to above into the format used by the specific host)
- Parsing of on-line response (verifying format and check fields)
- Communications with external host system
- Communications error handling
- Message integrity checks

External Communications will also be responsible for maintaining parameter sets directly relevant to each of the third party hosts it supports. These will consist of a wide range of details categorised as follows:

- addressing information for accessing third party hosts
- actions to take on interpretation of sent messages
- actions to take on interpretation of received messages



Guideline G18 – Configuring Integrated Systems

- rules to follow on communication failure
- protocol conversion rules and substitutions
- parameters related to the supported set of low-level communications drivers

External communications is also responsible for communicating with a software server, for downloading applications directly to the card (in, for example, a Multos or JavaCard environment) and for deleting such applications. This is distinct from the functions performed by Application Control, which manages the downloading of applications within the acceptance system ("Card Terminal Applications"). In this instance the system must conform to the specifications of the relevant scheme; at present Multos and Global Platform provide the only complete frameworks for this process.

Application Control

The main responsibility of Application Control is to manage the full set of valid Application Logic services, including applets where used. The service should also include control of the downloading process (where terminal applications are downloaded) or of the server functions for applets.

Specific functions will include:

- Application list updating
- Authentication key updating
- Application updating
- Application integrity checks

A significant proportion of responsibilities of this service are carried out at start-up or close down via the operating system and do not involve the operational interface via Business Logic.

All tasks will be determined by sets of parameter tables which this service must manage and maintain. Likewise, any application authentication information or public keys must be similarly held in secure storage. The parameter tables must contain the following information:

- Lists of stores/tills which have authority to use each version of an application
- Version number of each application supported
- Electrical and protocol standard supported by each application

-END OF DOCUMENT-