

nexo specification - Part 4

nexo-IS

JUNE 2021

DAVID HOARAU - DAVID.HOARAU@GALITT.COM

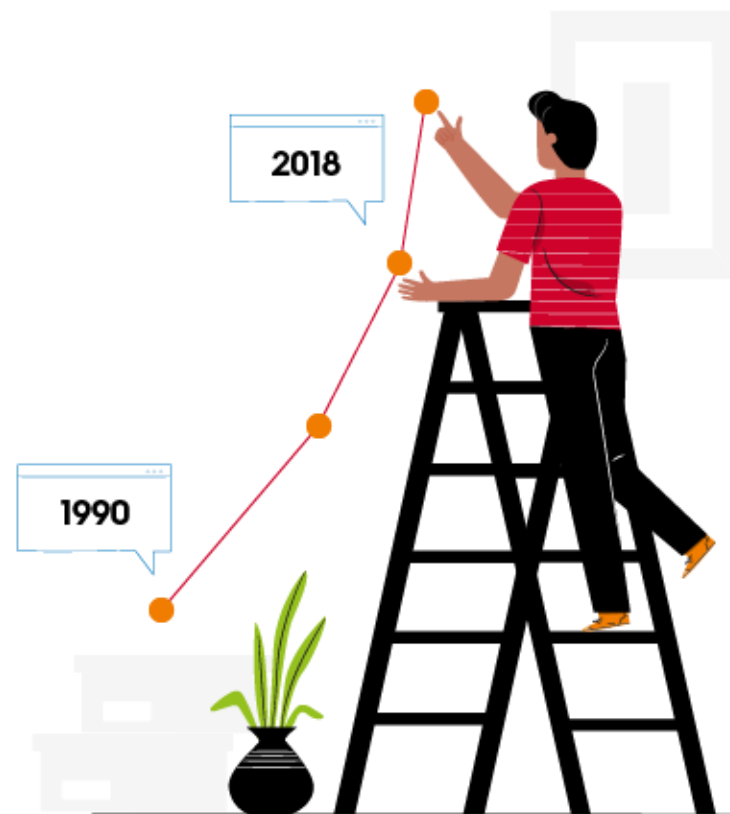
REF: 4- NEXO-IS – V1.03.PPTX

Passion for payments



Summary

- 01 Introduction of nexo-IS
- 02 HAP functionality
- 03 TMAP functionality
- 04 SCAP functionality



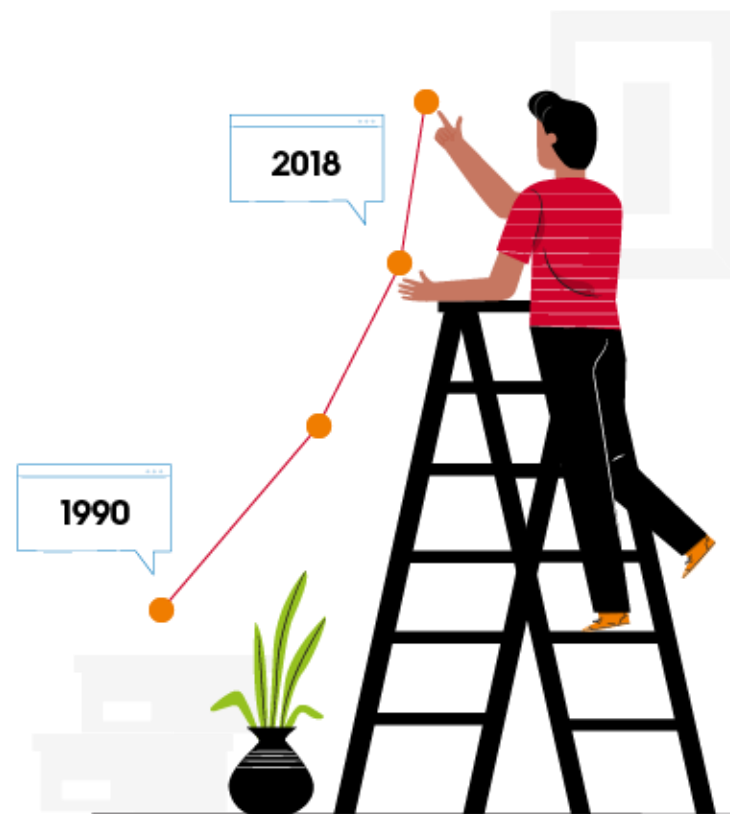
Summary

01 Introduction of nexo-IS

02 HAP functionality

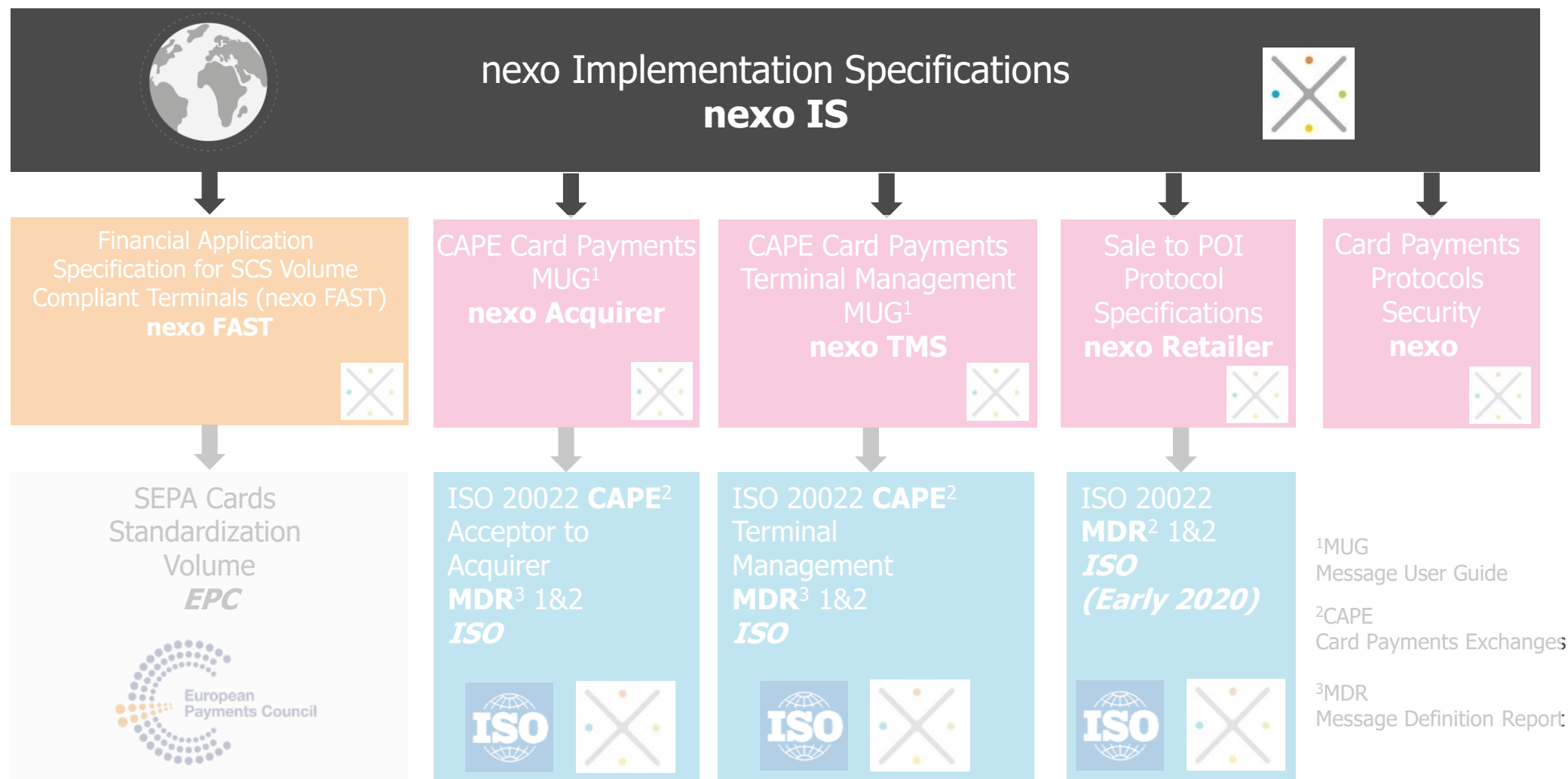
03 TMAP functionality

04 SCAP functionality



Foreword

Reference documents



Introduction of nexo IS

Scope of nexo IS



— The aim of this specification is:

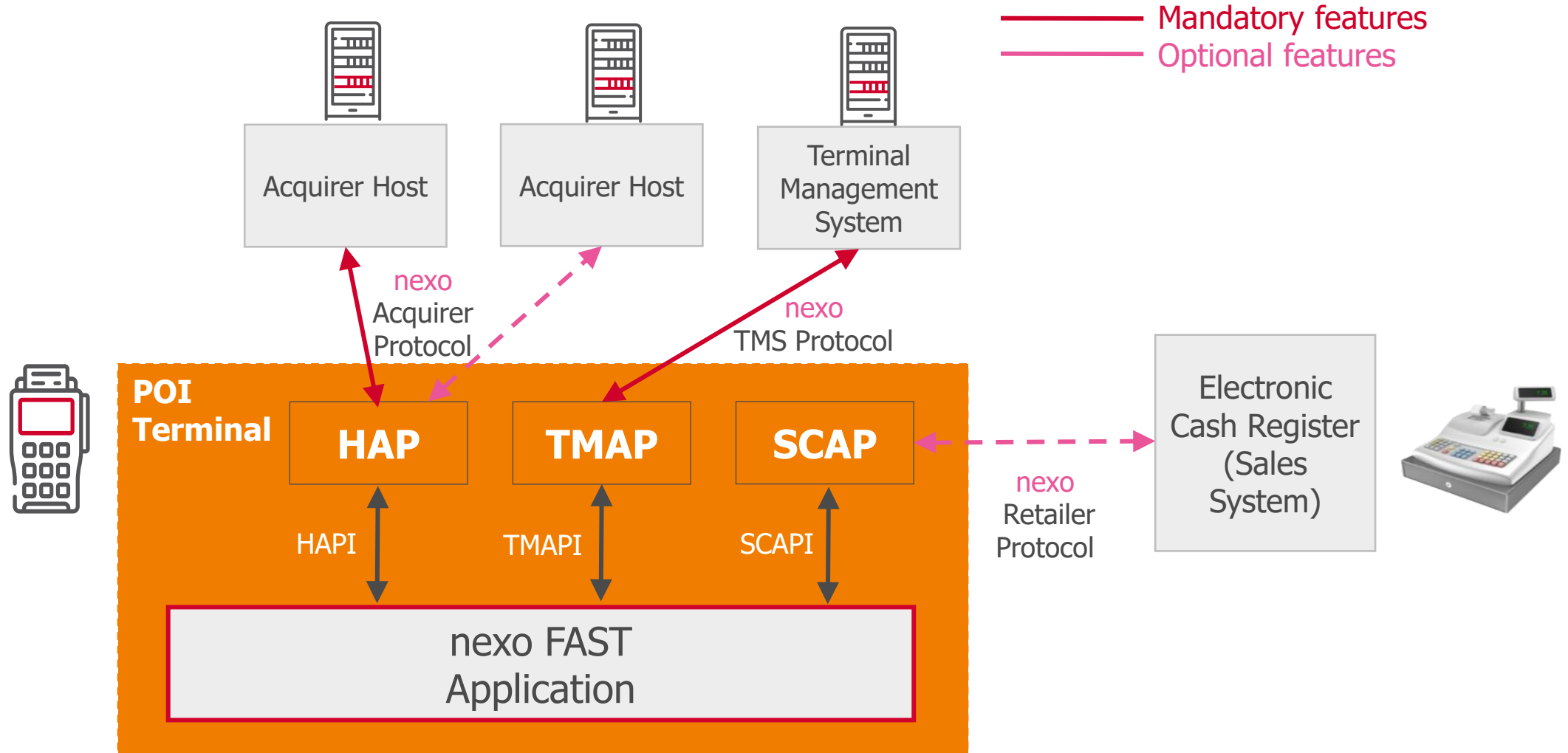
- └ to define in detail the **structure** and the **functions** of the **HAP** application
- └ to define an **interface** application for the **terminal management system (TMAP)**
- └ to define the **interaction** between the **sale system** and the dedicated interface (**SCAP**) using the nexo Retailer protocol

— Nexo IS main goal is to:

- └ To prove that POIs, developed on **nexo FAST** and **nexo Acquirer/TMS** systems may work **together**
- └ To prove the **interoperability** between all actors (POIs and Acquirers) implementing the nexo protocols

Introduction of nexo IS

Scope of nexo IS



Introduction of nexo IS

Scope of nexo IS



- A nexo IS **POI terminal** has to support
 - └ A nexo FAST application
 - └ HAP
 - └ TMAP
 - └ Connections to at least one Acquirer host
 - └ Connection to one TMS host for at least the purpose of nexo IS parameters download
 - └ Cryptographic mechanisms
 - └ SCAP (Optional)



- An **Acquirer** implementing the nexo IS specification has to support
 - └ The nexo Acquirer protocol
 - └ Cryptographic mechanisms
 - └ The TMS protocol (i.e, only if the acquirer is the terminal manager as well)
- A **Terminal Manager** implementing the nexo IS specification has to support
 - └ The TMS protocol
 - └ Cryptographic mechanisms

Introduction of nexo IS

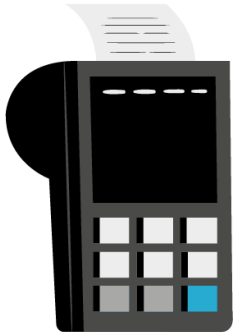
POI functional scope

— In scope:

- └ **Payment / Cancellation** (only if attended POI) / **Refund** (only if attended POI)

— Optional:

- └ Pre-Authorisation / Payment with Increased Amount or Cashback / Voice Authorisation / Deferred Payment / DCC* / No-Show* / Card Validity Check* / Quasi Cash* / Original Credit* / Instalment Payment* / Recurring Payment* / MoTo* / Choice of Application*, Cash advance*



— The vendor may decide:

- └ To develop an application for “Chip only” terminal
- └ To develop CL technology without the CT technology
- └ To support one or more of the CL kernels (2 to 7)
- └ The following combinations are possible:

- | | | | |
|--------------------------|--------------------------|--------------------------|-------------------------|
| ▪ Chip with contact EMV, | | | |
| ▪ Chip contactless EMV, | ▪ Chip with contact EMV, | ▪ Chip with contact EMV. | ▪ Chip contactless EMV. |
| ▪ Magnetic Stripe, | ▪ Chip contactless EMV. | | |
| ▪ Manual Entry. | | | |

Introduction of nexo IS

POI functional scope



Cardholder Verification Methods

- └ Online PIN,
- └ Encrypted offline PIN,
- └ Plain text offline PIN,
- └ NO CVM,
- └ Signature,
- └ CDCVM (Consumer Device CVM)

Card Authentication Methods

- └ SDA
- └ DDA
- └ CDA Mode1

POI Acquirer Protocol Modes

- └ Mode 1 (Capture by completion)
- └ Mode 2 (Capture by batch)
- └ Mode 3 (Capture by authorisation)

TMS Protocol support

- └ StatusReport/ManagementPlanReplacement exchange
- └ StatusReport/AcceptorConfigurationUpdate exchange
- └ Key download
- └ TMS Delegation

Cryptographic mechanisms

- └ DUKPT/UKPT (for PIN/CardData/MAC)
- └ Triple-DES and AES128 (for PIN/CardData)
- └ Retail-CBC-MAC with SHA-256 (or MAC)
- └ Key download via TMS protocol

Introduction of nexo IS

Acquirer functional scope



ACQ

- **In scope:**

- └ **Payment / Cancellation / Refund**

- **Optional:**

- └ Pre-Authorisation / Payment with Increased Amount or Cashback / Voice Authorisation / Deferred Payment / DCC* / No-Show* / Card Validity Check* / Quasi Cash* / Original Credit* / Instalment Payment* / Recurring Payment* / MoTo* / Choice of Application*/Cash Advance*

- **Acceptance Technology:**

- Chip with contact EMV,
- Chip contactless EMV,
- Magnetic Stripe,
- Manual Entry.
- Chip with contact EMV,
- Chip contactless EMV.
- Chip with contact EMV.
- Chip contactless EMV.

Introduction of nexo IS

Acquirer functional scope



ACQ

Cardholder Verification Methods

- └ Online PIN,
- └ Encrypted offline PIN,
- └ Plain text offline PIN,
- └ NO CVM,
- └ Signature,
- └ CDCVM (Consumer Device CVM)

POI Acquirer Protocol Modes

- └ Mode 1 (Capture by completion)
- └ Mode 2 (Capture by batch)
- └ Mode 3 (Capture by authorisation)

Cryptographic mechanisms

- └ DUKPT/UKPT (for PIN/CardData/MAC)
- └ Triple-DES and AES128 (for PIN/CardData)
- └ Retail-CBC-MAC with SHA-256 (or MAC)

Introduction of nexo IS

Terminal management functional scope

- The Terminal Management System carries out the POI management, and must support the functionalities detailed below



TMS

Cryptographic mechanisms

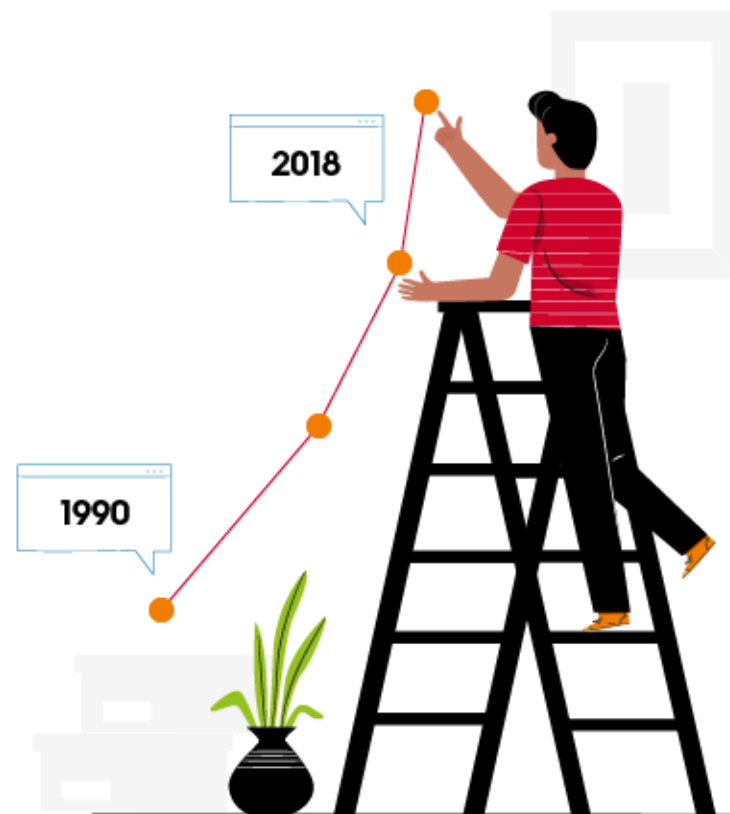
- └ DUKPT/UKPT (as derivation for MAC)
- └ Retail-CBC-MAC with SHA-256 (or MAC)
- └ Key download via TMS protocol

TMS Protocol support

- └ StatusReport/ManagementPlanReplacement exchange
- └ StatusReport/AcceptorConfigurationUpdate exchange
- └ Key download
- └ TMS Delegation

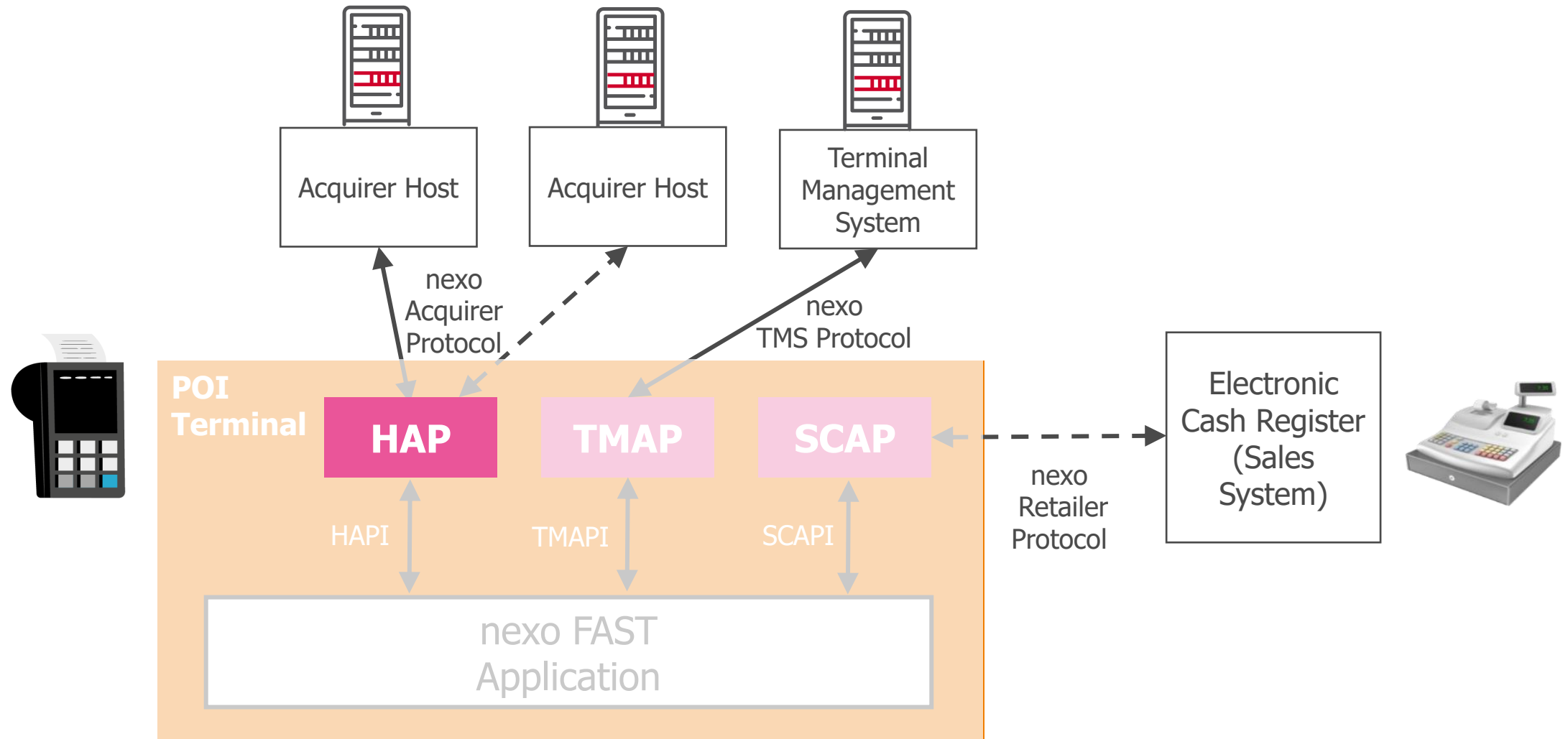
Summary

- 01 Introduction of nexo-IS
- 02 **HAP functionality**
- 03 TMAP functionality
- 04 SCAP functionality



HAP functionality

Scope



HAP functionality

Scope

- **HAP is the application responsible for handling the acquirer protocol and has access to all nexo FAST application data**
- **HAP manages the following:**
 - └ Conversion of data elements exchanged between nexo FAST and nexo Acquirer Protocol
 - └ The communication between the POI application and the host using the nexo Acquirer Protocol



HAP functionality

HAP status

-  — Called during the **Start-up** of the Financial Application
-  — Shall process mandatory tasks as
 - └ **Check of the availability of mandatory data** required for the financial application and the acquirer protocol
 - └ **Transmission** respectively retransmission of any outstanding pending **transactions**
 - └ Transaction **recovery** because of power failure
-  — Other actions could be processed if triggered manually or by configuration
 - └ **Diagnostic exchange** if triggered manually by the merchant
 - └ **Batch Transfer** if required
 - └ **Reconciliation** if triggered manually by the merchant or configured by TMS

HAP functionality

HAP online approval request

- Checks the **Selected Service** to build the **corresponding** nexo Protocol **message**
- Checks that all mandatory data needed for the protocol are available (Data elements not provided by nexo FAST are generated by HAP)
- Build Request Message : **AuthorisationRequest** or **CancellationRequest**
 - └ Build Message **Header**
 - └ Build Message **Body**
 - └ Build Message **Security Trailer**
- **Send message** to the recipient : according to the **Acquirer Number**



HAP functionality

HAP online approval request

— Receive & Parse Response Message

└ Data elements to be verified

ResponseToAuthorisation.Response

Action.ActionType

Action.MessageToPresent.MessageContent

Action.MessageToPresent.MessageDestination

Data in Response Message		Data set by HAP		
ResponseToAuthorisation.Response	TransactionResponse.Action.ActionType	Transaction Result	ARC ⁴	Nok Reason
Approved	Equal to "PrintMessage", "DisplayMessage", OR no Action present	APPROVED	"00"	-
Approved	Different than "DisplayMessage", "PrintMessage"	ABORTED	-	DATA ERROR
Declined	Referral and optional "DisplayMessage", "PrintMessage"	VOICE AUTHORISATION	"02"	-
Declined	Message element absent, "DisplayMessage" and/or "PrintMessage"	DECLINED	"05"	-



- └ If an error is present HAP shall send an AcceptorCompletionAdvice or an AcceptorCancellationAdvice message during "HAP transaction finalisation" with message element **Reversal** set to "True"

- If the data element TransactionResponse.**TMSTrigger** is present in the response message, HAP shall take into account the instructions to call the TMS (TMSContactLevel : **AsSoonAsPossible, Critical or DateTime**)

HAP functionality

HAP data storage



- The **storage** of the transaction data at **Transaction Completion**, necessary for
 - └ the **capture** of the transaction by an **AcceptorCompletionAdvice** message if configured by TMS
 - └ the **capture** of the transaction by a **BatchTransfer** if configured by TMS
 - └ the **cancellation** of the transaction



- For **batch capture**, transactions kept until a valid BatchTransferResponse with **RemoveDataSet** flag set to **true**



- For **authorisation** and **completion capture**, transactions kept until a valid **ReconciliationResponse**

HAP functionality

HAP transaction finalisation



- **Performed at Transaction Completion**

- **Advice** handling for completion and/or reversal

- For reversal: Transaction.Reversal is set to "True" and Transaction.FailureReason is present

- For approved Transaction: Transaction.TransactionSuccess is set to "True"

- Otherwise the Transaction.TransactionSuccess is set to "False"

- **Capture** of the data



- Build Request Message : **AcceptorCompletionAdvice** or **AcceptorCancellationAdvice**



- Sends advice message according to the **Acquirer Number**

HAP Functionality

HAP transaction finalisation



— Receives Response Message

- └ checks the consistency of the message syntax and the MAC

If error, retransmit the advice message as soon as possible

Otherwise updates the appropriate terminal transaction data elements



— Updates **Finalization Flag** to “Finalized”

- └ when a transaction completion was required, and the completion was performed successfully
- └ when no completion was required for the transaction

HAP functionality

HAP search transactions



- The nexo FAST application calls this function to request a search for, and the retrieval of data of previous transaction(s) in the log



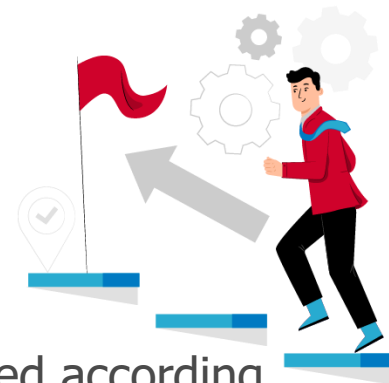
- The function is used in the following Services
 - └ Deferred Payment Completion
 - └ Cancellation
 - └ Pre-Authorisation
 - └ Update Pre-Authorisation
 - └ Payment Completion
 - └ No-Show
 - └ Voice Authorisation



- Additionally, this function is used by Kernel E to check the previous transactions to prevent split sales.

HAP fonctionnalité

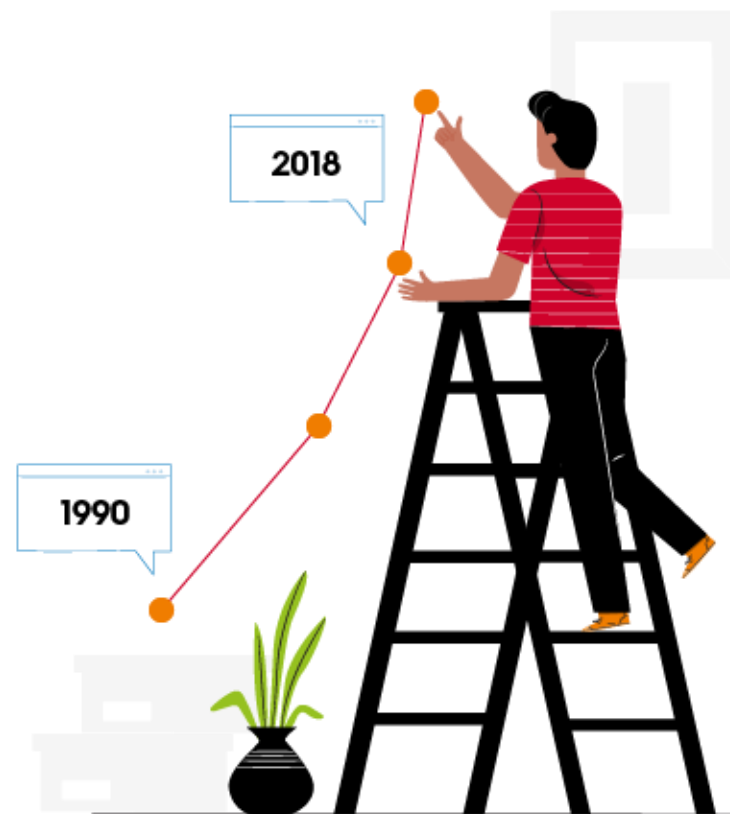
Batch processing



- The POI generates the BatchTransfer message/file containing all transactions to be captured according to the TMS configuration (Offline and Online **ExchangePolicy**)
 - └ NumberLimit
 - └ TotalLimit
 - └ Cyclic
- The BatchTransfer may also be **initiated by the merchant** using a local administration interface of the POI
- Build and Fill in DataSet information (CreationDateTime, DataSetIdentification, TransactionTotals...)
- Send BatchTransfer message
- Receive and parse BatchTransferResponse
 - └ If the validation of the response message fails, HAP shall keep all transactions
 - └ Otherwise, if RemoveDataSet is set to true then HAP deletes the logs

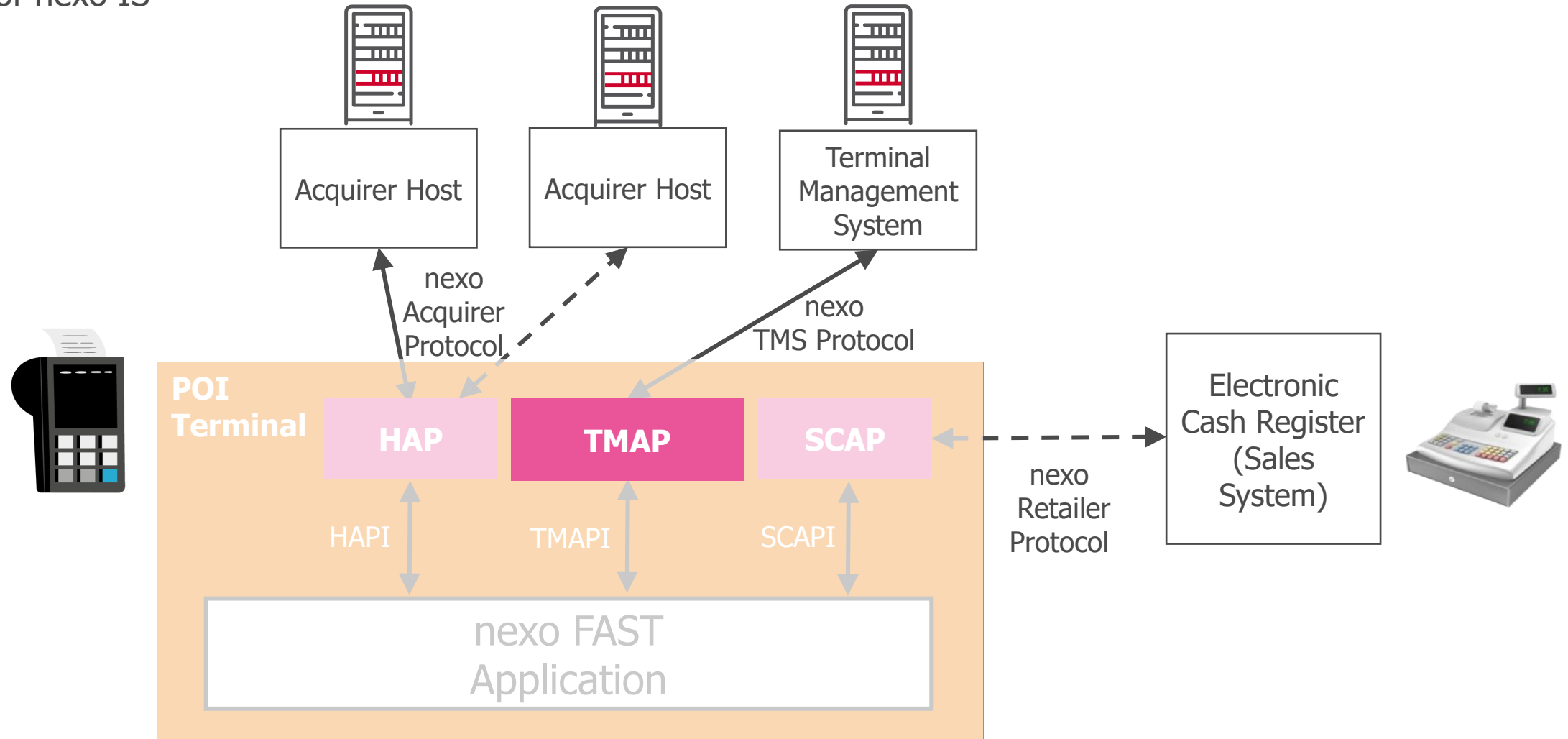
Summary

- 01 Introduction of nexo-IS
- 02 HAP functionality
- 03 **TMAP functionality**
- 04 SCAP functionality



Introduction of nexo IS

Scope of nexo IS



TMAP functionality

Scope

- Handles the protocol used for **maintenance** and **administrative** functionality
- Manages
 - └ nexo FAST configuration data elements provided by TMS
 - └ nexo Acquirer Protocol configuration data elements provided by TMS
 - └ Communication between POI application and TMS
 - └ Consolidation of the templates received from different Terminal Managers.
 - └ Checking of rights for a delegated TM



TMAP functionality

Perform terminal management session



- Performed during the **Idle State**
- While running a TMS session **no service shall be started**
- Triggers the TMS session depending on **Call TMS**
 - └ ASAP or Critical: immediately
 - └ DateTime: at the date and time specified in the Action.TimeCondition data structure of the ManagementPlan



- If **Terminal Error Indicator** is set to "TRUE" and **Nok Reason** is set to "CONFIGURATION ERROR", sends description of the missing or incorrect parameters in the subsequent StatusReport
- At the end of the TMS session, TMAP shall
 - └ **Check the received configuration**
 - └ Ensure that all mandatory data is available
 - └ Generate a configuration report, if applicable
 - └ Set the element **Call TMS** to "None" if the TMS has been performed successfully

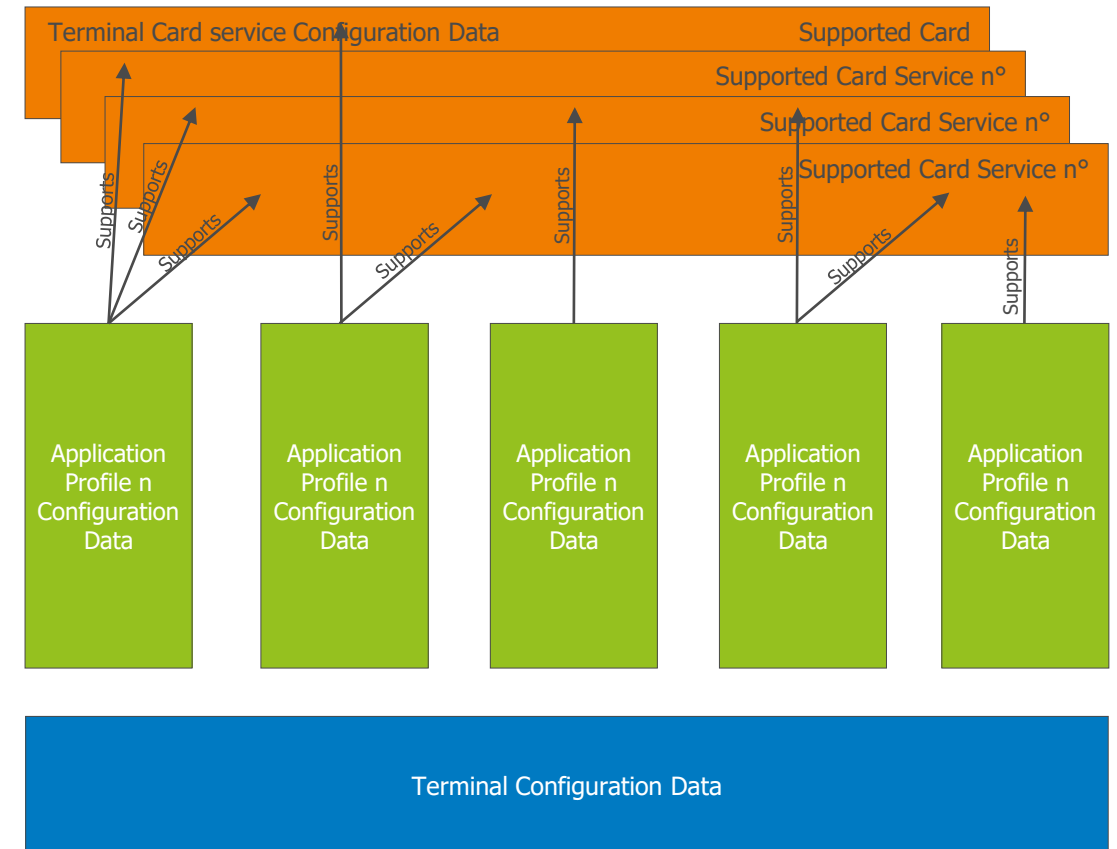


TMAP functionality

nexo FAST configuration parameters

Logical parameters are classified into three groups

- └ **Terminal configuration data**
- └ **Terminal card service configuration data**
- └ **Application profile configuration data**

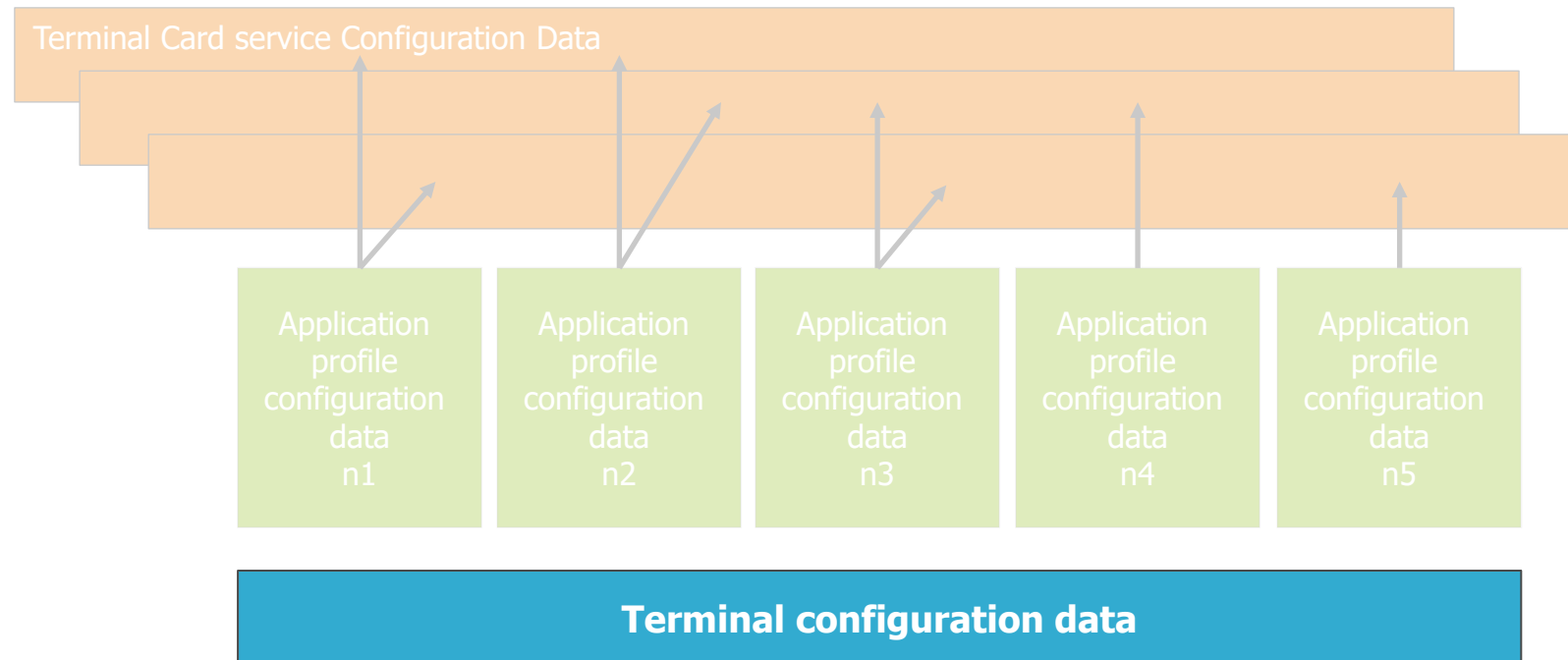


TMAP functionality

nexo FAST configuration parameters

— Terminal configuration data

- └ There is only one occurrence of these data elements by terminal
- └ This configuration data includes operating data, security data (certificate of public keys)
- └ *Examples : Terminal capabilities, Application Profile AID, Supported Services, ...*

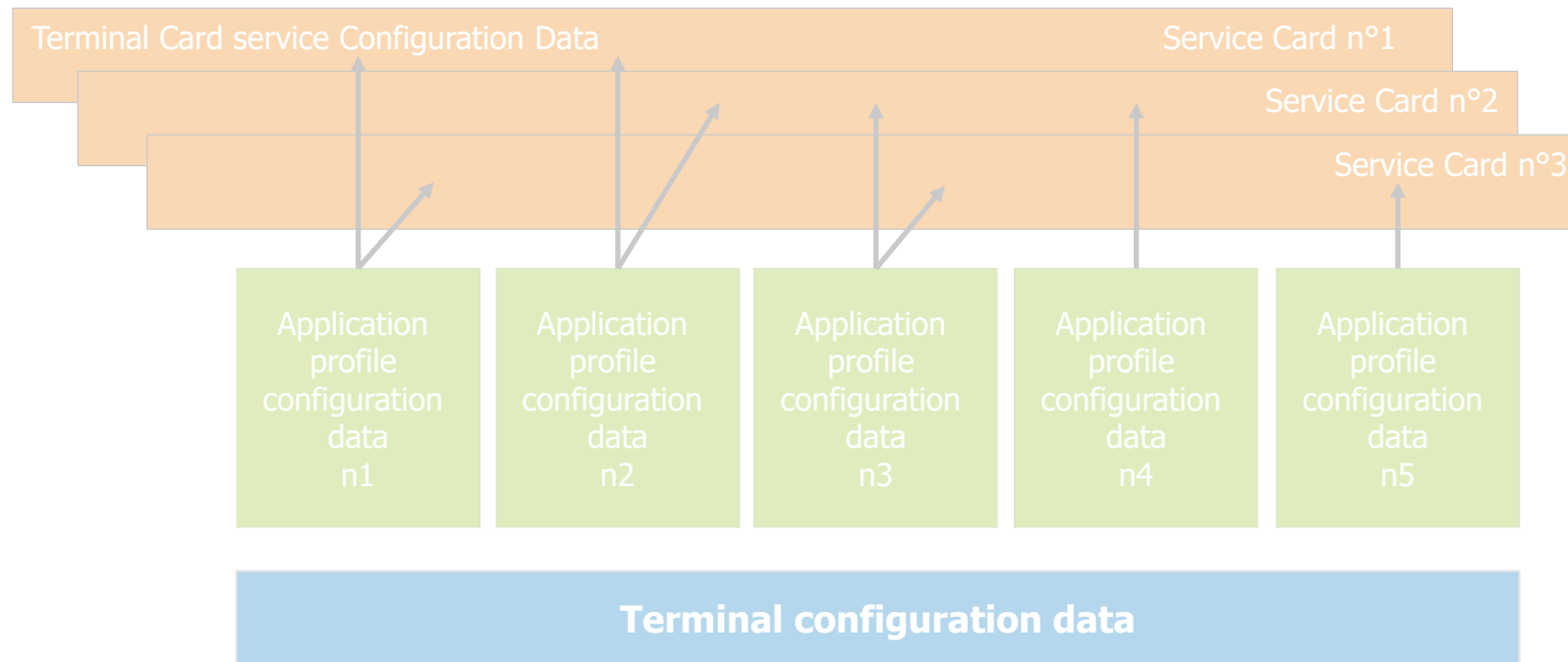


TMAP functionality

nexo FAST configuration parameters

— Terminal Card service configuration data

- └ There must be as many occurrences as there are services possibly supported by the nexo FAST application
- └ Examples : *Service Settings, Allowed Service Start Events, Minimum Service Start Conditions, ...*
- └ Correspondence with the spec nexo FAST : Card Service Processing



TMAP functionality

Application profile

Selected parameter set according to the card and used in the nexo FAST application



— Objective

- └ Allows you to restrict the services of a POI according to the card with which it interacts



— Rules

- └ Main Application Profile Selection Criteria: AID of the map and map service selected
- └ One acquirer and one AID per application profile
- └ The chosen application profile remains the same for an entire transaction
- └ Updated by the TMS



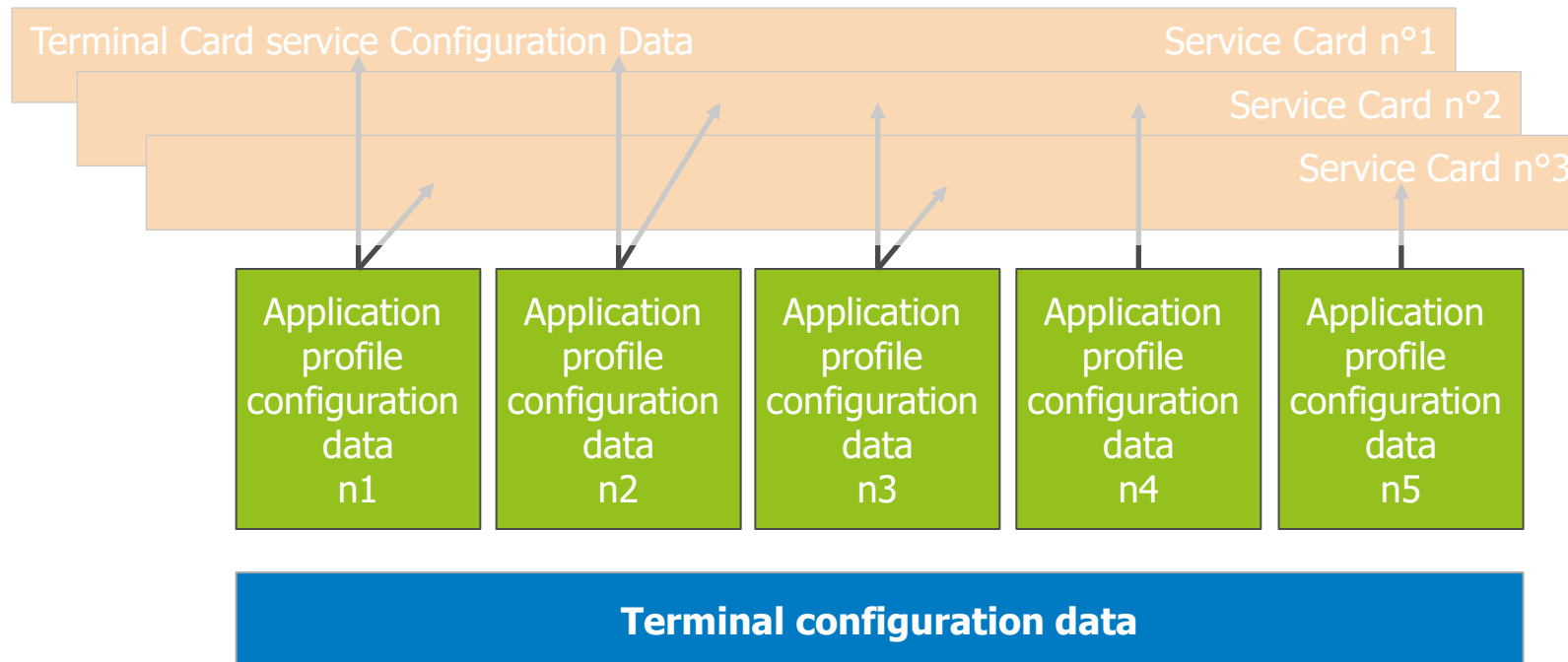
— There may be fairly general profiles, and more specific profiles (at a given AID corresponds a specific behavior)

TMAP functionality

nexo FAST configuration parameters

— Application profile configuration data

- Multiple occurrences of application profile configuration data elements might be present in the nexo FAST depending on acquirer parameters downloading
- At least one set of application profile configuration data element must be present
- Examples : *Application Profile Settings, Terminal Capabilities, Terminal Floor Limit, ...*



TMAP functionality

Structure of data elements table

Template	Label table	Content	Occurence
E0	Permanent terminal specific data	Merchant specific terminal data, hardware, ... (<i>Terminal indentification, Comman Key Labels, ...</i>)	1..1
E1	Terminal specific data	Terminal specific settings (<i>Terminal Capabilities, Terminal Country Code, Terminal Type, Terminal Transaction Currency Code, Default Card Service</i>)	1..1
E2	Application profile selection table	Table listing data associated with AID (Supported Services, Applicaiothn profile number), used in the application profile number search process from AID	1..N
E3	Certification authority public key table	List of certification keys for each scheme	1..N
E4	Service settings table	For each service (payment, cancellation, refund...) mode of treatment to be performed by the terminal, triggering event (Chip, Magstripe, Manual,...)	1..N



TMAP functionality

Structure of data elements tables

Template	Label table	Content	Occurence
E5	Terminal list of AID	List of AIDs supported by the terminal, associated with indicator (<i>Application Selection Indicator</i>)	1..N
E6	Application profile parameters	For each application profile, list of application parameters transmitted by the acquirer (limit amounts, merchant IDs, supported services, etc)	1..N
E7	Terminal list of BID	List of products card (BID) supported by the terminal. Used for application selection when playing track or manual input	1..N
E8	Application profiles selection table non chip	Table listing the data associated with the BID (supported services, acquirer identifier, ...) used in the process of searching the application profile number from the BID	1..N
E9	Terminal exception file	Black list of PANs (lost, stolen,...)	1..N



TMAP functionality

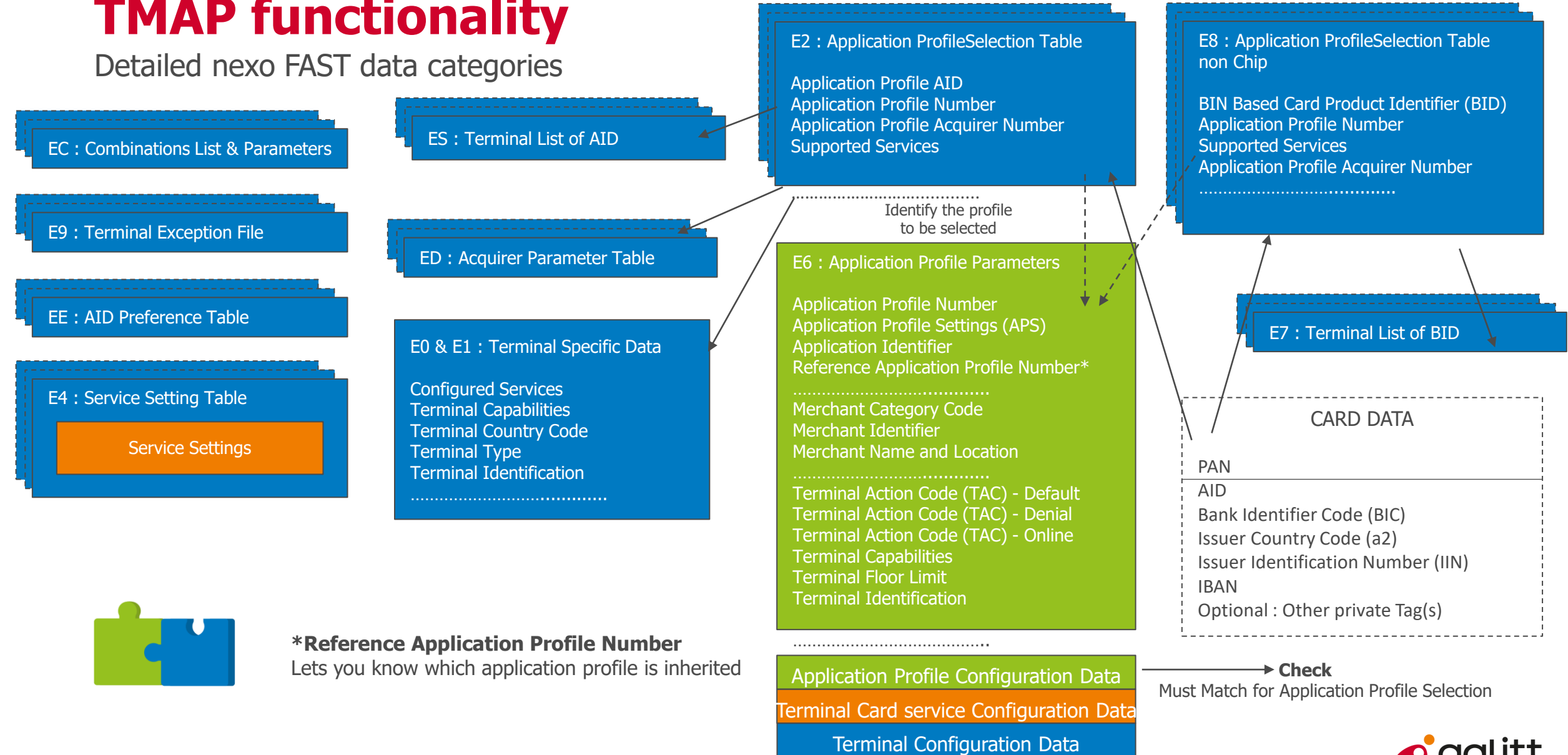
Structure of data elements tables



Template	Label table	Content	Occurence
EA (Removed nIS v5)	Limit set list	List of contactless limits for Kernel C-3 (limit amounts ...)	1..N
EB	Default kernal ID per AID	Table designating the Kernel to use by default according to the AID (case of the card, in contact)	1..N
EC	Combinations list & parameters table	Table linking an AID-Kernel couple to one of more services. Also contains settings for contactless (if supported)	1..N
ED	Acquirer parameter table	Concatentation of the acquirer parameter table for each acquirer supported by the terminal	1..N
EE	AID preference table	List of priority AIDs in the case of multi-application cards	1..N
F0	EEA process table for template	European Economic Area specific	
F1	EEA-only list of AID for template	European Economic Area specific	
F2 (Added nIS v5)	DCC Excluded BIN List		

TMAP functionality

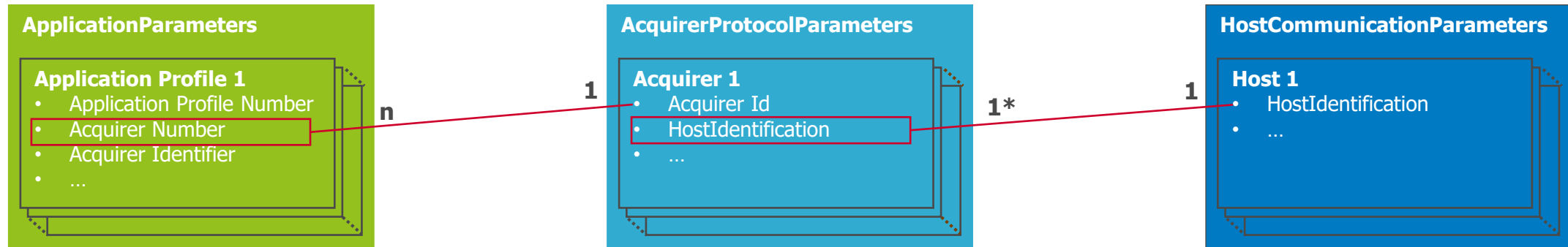
Detailed nexo FAST data categories



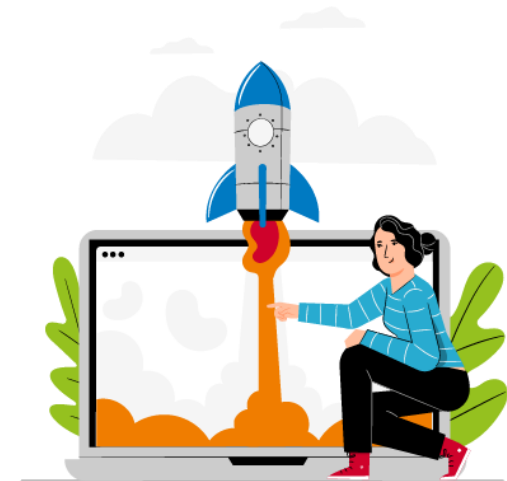
***Reference Application Profile Number**
Lets you know which application profile is inherited

TMAP functionality

Detailed nexo FAST data categories

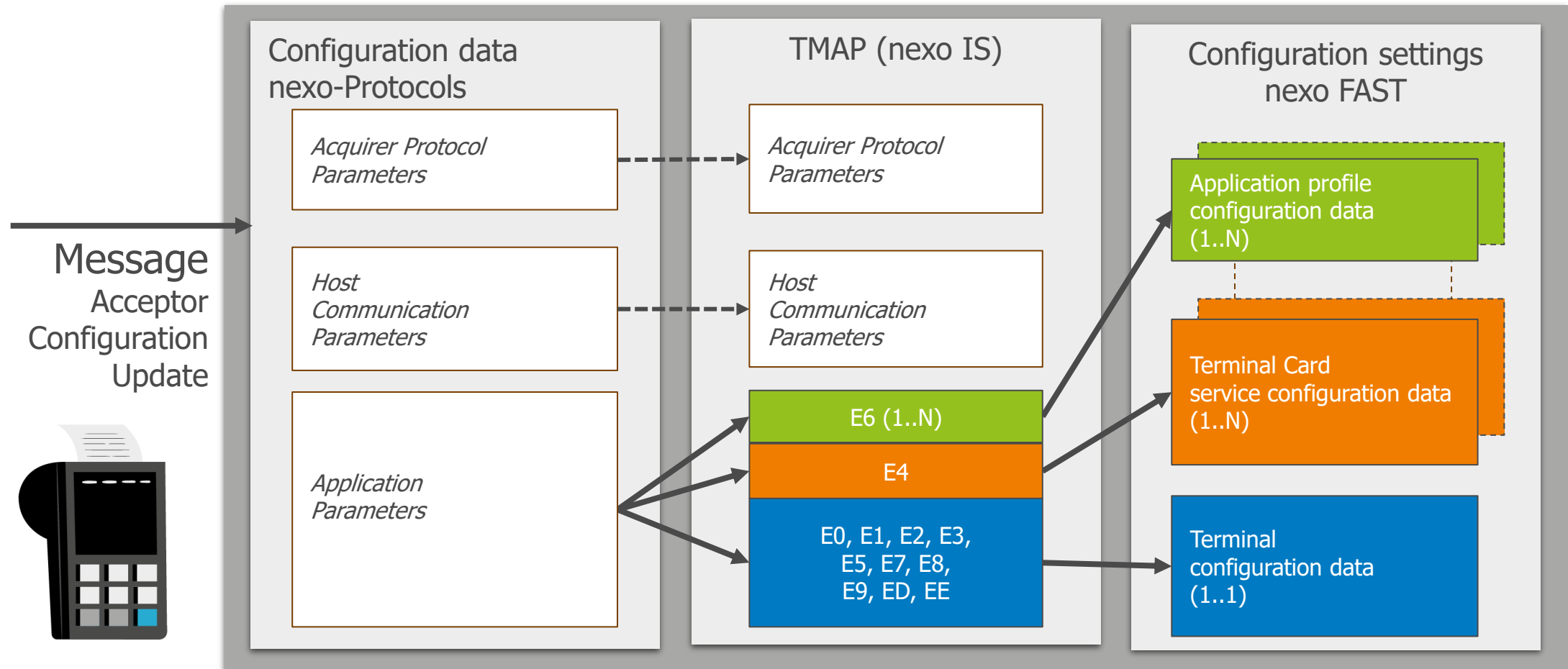


* It is technically possible to associate several acquirers to the same 'Host'



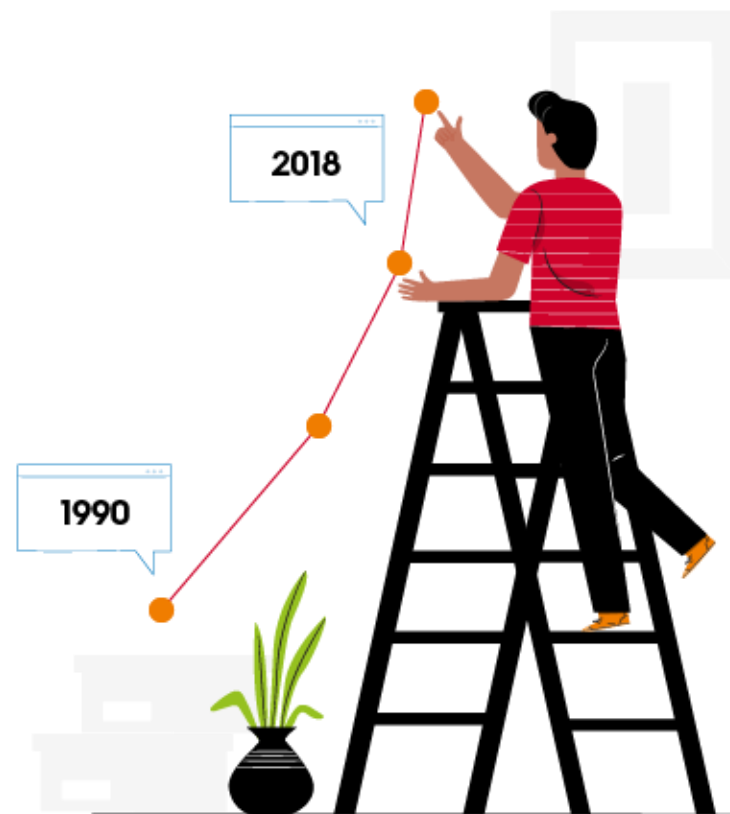
TMAP functionality

Correspondence of the nexo parameters



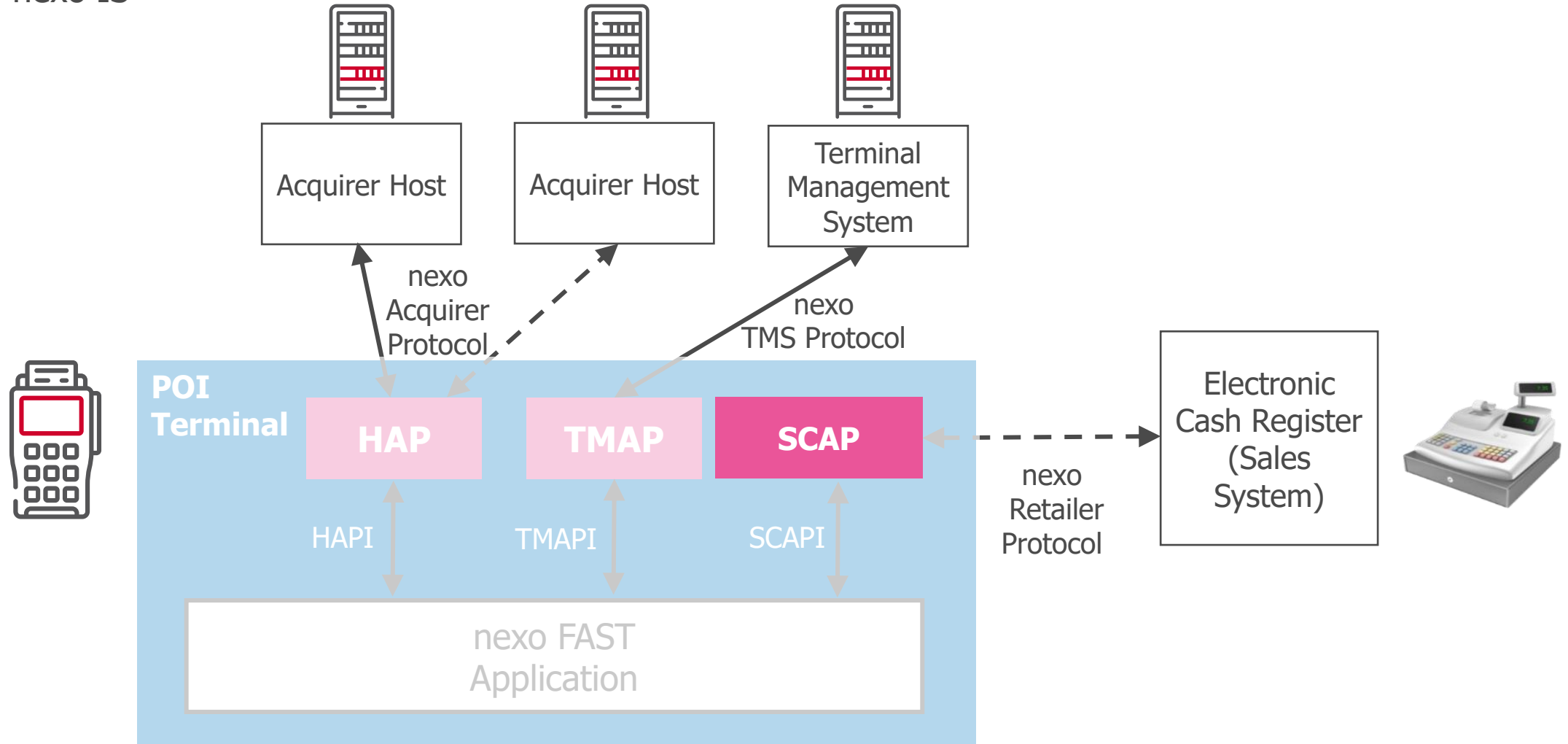
Summary

- 01 Introduction of nexo-IS
- 02 HAP functionality
- 03 TMAP functionality
- 04 **SCAP functionality**



Introduction of nexo IS

Scope of nexo IS



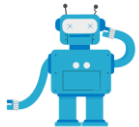
SCAP functionality

Interface with the sale system



- For the functional scope of the nexo IS payment system, the support of the retailer protocol is optional.
- The following Retailer messages must be supported:

- └ **Diagnosis** to verify the status of the Acquirer host connectivity
- └ **Standard Payment** for the processing of various Payment services
- └ **Reversal** for payment cancellation, if cancellation is supported
- └ **TransactionStatus**, **Abort** and **Event** for error handling
- └ **Display** for the output of cardholder and merchant messages when these interfaces are available on the Sale System
- └ **Print** for the print-out of cardholder and merchant receipts at the sales system's printer
- └ **Reconciliation** to exchange totals with the Sale system, and to initiate reconciliation with one or several acquirers



- Complementary optional services of Retailer may be needed to provide the functions required by the Sale system:
 - └ **Login** and **Logout** for the session management of the Sale System. A session could represent a shift of the cashier, a business day or any other period at the convenience of the merchant
 - └ The **interface** between nexo FAST and SCAP is called SCAPI