



13<sup>th</sup> - 17<sup>th</sup> May 2019 – Kuala Lumpur

# Fraud Model Development and Deployment in SAS FM

Session 2: Message Layouts in SAS FM

# Message Layouts in SAS FM

## Topics

- Basic design
- The SMH segment and message construction
- How to read the API document?
- Review of common segments
- API mapping
- Flexible API
- TDR and consortium



# Message Layouts in SAS FM

## Basic Design

# Message Layouts in SAS FM

## Basic Design

- SAS FM provides a standard message layout specification for constructing all messages that need to be passed into the system.
  - Sometimes referred to as the SAS FM API
  - Modular in nature
- We define a “message” as an information unit flowing into the system.
  - Every event is constructed as a message and fed into the system.
  - Monetary examples: Debit card purchase, mobile deposit, funds transfer.
  - Non-monetary examples: Balance enquiry, online banking logon, address change, account master information, fraud feedback.

# Message Layouts in SAS FM

## Basic Design

- Messages are constructed as an ordered set of segments; segments are collection of fields in a given order
  - Segments are denoted / addressed by a 3 letter acronym
- Depending on the nature of transaction, one can select the appropriate segments to construct the message
- Every message starts with the SMH (System Message Header) segment:
  - Contains a list of fields whose values determine the remaining segment order in the message
  - Different values for these header fields are associated with different segments
- The different segments can be combined in numerous ways to define a multitude of message types
  - Software currently allows about 15,000; most customers use < 10

# Message Layouts in SAS FM

## System Message Header Segment (SMH)

Field	Description	Possible Values
smh_tran_type	Transaction Type	<b>TRX: Financial Transaction</b> CMD: System Transaction MON: System Monitor Transaction DMA: Distributed Message ACK
smh_cust_type	Customer Type	B: Business I: Individual N: Not Applicable
smh_acct_type	Account Type	CC: Credit Card Account CS: Checking/Saving Account SL: Secured Loan LC: Line of Credit BR: Brokerage Account MA: Merchant Account NA: Not Applicable
smh_authenticate_mtd	Authentication Method Type	CP: Chip Card Related Authentication Method CD: Non-Chip Card Authentication Method NC: Non-Card Related Authentication Method BT: Brokerage Transaction UK: Unknown NA: Not Applicable

# Message Layouts in SAS FM

## System Message Header Segment (SMH)

Field	Description	Possible Values
smh_channel_type	Channel Type	C: Payment Card at Card Reader Terminal (including online purchase and ATM) D: Payment Card or Number with Online Details and Device Fingerprint Information E: Payment Card or Number with Online Details O: Online Banking (internet, mobile phone) W: Online Banking with device fingerprint information P: Phone Banking H: Self Bank Branch M: Correspondence(for non-mon and check deposit) B: Bank Processing(include bank initiated non-mon maintenance, ACH debit, EFT processing) F: Financial Consultant R: Other S: Merchant - Acquirer Processing with Device Fingerprint T: Merchant - Acquirer Processing U: Unknown N: Not Applicable

# Message Layouts in SAS FM

## System Message Header Segment (SMH)

Field	Description	Possible Values
smh_activity_type	Activity Component Type	CA: Payment Card Authorization CP: Payment Card Postings SH: Bill Payment/Fund Transfer Scheduling and Cancellation BF: Bill Payment/Fund Transfer to Third Party or Self Account CK: Check Payment CW: Over-the-Counter Cash Withdraw DP: Deposit/Deposit Reversal AB: Batch ACH submission header/control record AE: Batch ACH submission entry record NM: Non-Monetary activities MI: Individual Customer Master File MB: Business Entity Customer Master File MR: Credit Card Account Master File MD: Checking/Saving Account Master File MS: Secured Loan Account Master File ML: Line of Credit Account Master File MC: Payment Card Master File MO: Online Banking User Master File ME: Staff ID Master File MM: Merchant MasterFile MT: Terminal MasterFile FM: Financial Transaction at Merchant UK: Unknown



# Message Layouts in SAS FM

## System Message Header Segment (SMH)

Field	Description	Possible Values
smh_activity_detail1	Additional segments to include if necessary to describe the activity in detail	NAP Not Applicable
		DNA Non-monetary Address Change
		DNP Non-monetary Phone Change
		DNC Non-monetary Credit Limit Change
		DNU Non-monetary Other Change
		DNR Non-monetary Reissue
		DNS Non-monetary Sensitive Data
		DNO Non-monetary Card Activation
		DNX Non-monetary Entity Transfer/Purge
		DCB Non-monetary Credit Bureau Info Update
		DCC Credit Card Account Cycle Cut data
		DCQ Cheque Detail Data
		DBA Business Payment Administration Data
		DMX Extra Modeling Data
		DEE Extra Entity Segment
		DPD Posting Details
		DPP New Payee or Address Book Item
		DUA User Authorization Request
		DUS On Us Transaction Details
		DCH Chargeback Details
smh_activity_detail2		<i>General purpose segments</i>
smh_activity_detail3		



# Message Layouts in SAS FM

## Message Construction

# Message Layouts in SAS FM

## Message Structure

- Messages are essentially created as packed byte streams
  - Individual fields are parsed by the ODE

Header	Common		Message Specific	Optional	Common			
SMH	RRR	RQO	Segments based on SMH header values	Activity detail segments (up to 3)	RUA	ROB	RDK	RUR

# Message Layouts in SAS FM

## Segments Associated with *SMH\_ACCT\_TYPE*

SMH_ACCT_TYPE	Associated Segments
CS	AQO, AQD
CC	AQO, AQC
SL	AQO, AQS
LC	AQO, AQL
BR	AQO, AQB, AQR
MA	AQM
NA	-

# Message Layouts in SAS FM

## Segments Associated with *SMH\_AUTHENTICATE\_MTD*

SMH_AUTHENTICATE_MTD	Associated Segments
CP	UCM, UCC
CD	UCM
NC	UNM
BT	UBT
UK	-
NA	-

# Message Layouts in SAS FM

## Segments Associated with *SMH\_CHANNEL\_TYPE*

SMH_CHANNEL_TYPE	Associated Segments
C	HQO, HCT
D	HQO, HCT, HOB, HDF
E	HQO, HCT, HOB
O	HQO, HOB
W	HQO, HOB, HDF
P	HQO, HPB
H	HQO, HBB
M	HQO, HCP

SMH_CHANNEL_TYPE	Associated Segments
B	HQO, HBP
R	HQO, HOT
F	HFC
T	HQM
S	HDF, HWM
U	HQO
N	-

# Message Layouts in SAS FM

## Segments Associated with *SMH\_ACTIVITY\_TYPE*

SMH_ACTIVITY_TYPE	Associated Segments
CA	TCA
CP	TCP
FM	TCM
SH	TSH, TPP
BF	TBT, TPP
CK	TCK, TPP
DP	TDP, TPP
CW	TCW, TPP
AB	TAQ, TAB
AE	TAQ, TAE, TPP
NM	TNG
MI	TMI
MB	TMB

SMH_ACTIVITY_TYPE	Associated Segments
MR	TMR
MD	TMD
MS	TMS
ML	TML
MC	TMC
MO	TMO
MP	TMP
ME	TME
ST	TST, TSR
MM	TMM
MT	TMT
UK	-

# Message Layouts in SAS FM

## Segments Associated with Other Header Fields

- Currently SMH\_CUST\_TYPE has only one segment associated with it
  - XQO for values of I (individual) or B (business)
  - No segment for value of N
- Segments associated with activity details 1 – 3 are the values given in the dictionary itself
  - i.e. smh\_activity\_detail1 = DMX corresponds to the DMX segment



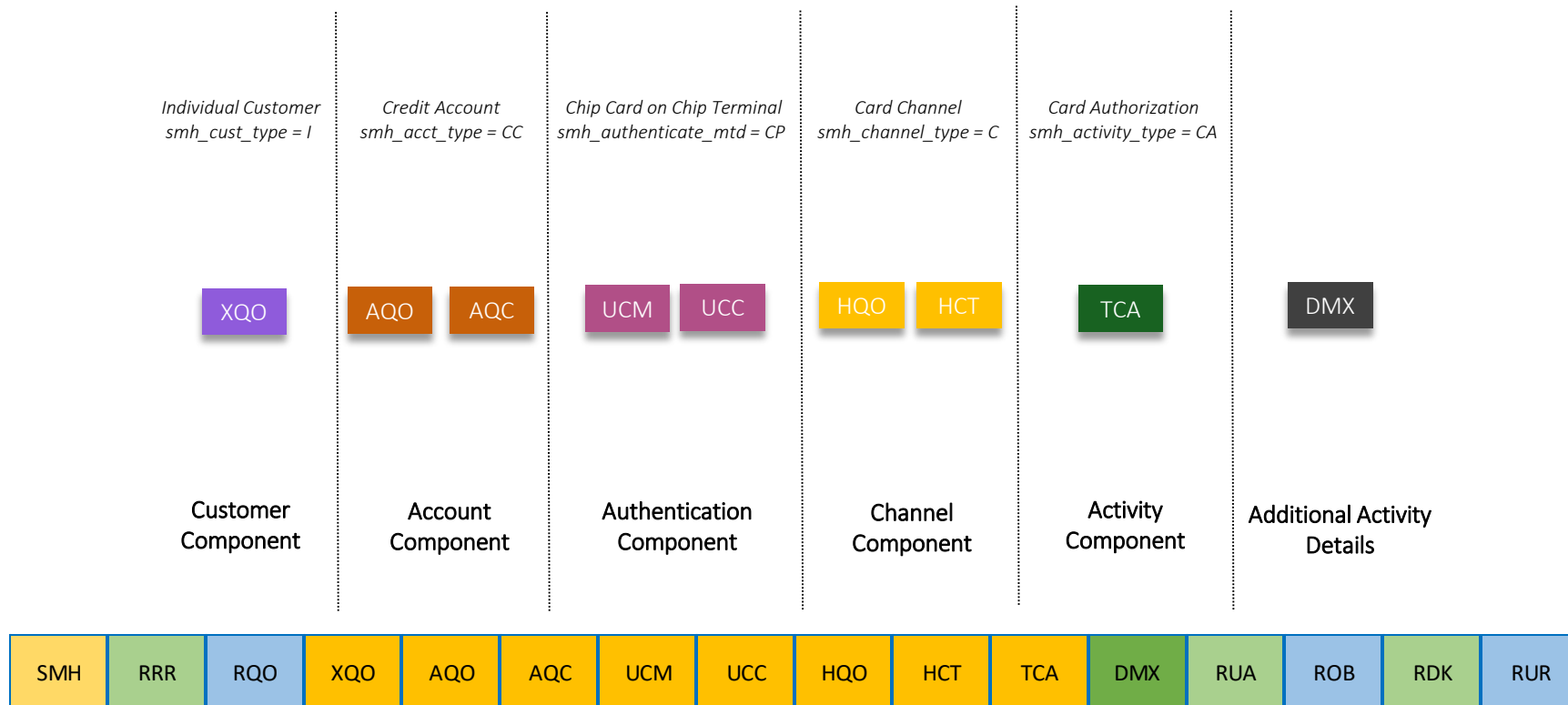
# Message Layouts in SAS FM

## Common Segments

- **RQO** contains date and time information common to every message type
- **RRR** is the 'return segment'. Contains fields for:
  - Model return values
  - Final decision related outcomes set by the rules
  - Any processing outcomes by the system
  - Should not be populated with values in the incoming message
- **RUA** and **RUR** are general purpose segments to be used by the rule writers
  - **RUA** is a read-only field and **RUR** is a writeable field (rules can assign values)
  - In theory, values can be changed at will and therefore should not be used by the models
  - However, in practice, the **RUA** segment is heavily used as a general purpose segment due to space constraints and utilized by the model
    - Should make sure the customer is aware of this
- **RDK** is designed to be utilized by the model for CPP stamping
  - Outside the scope of our discussion
- **ROB** (one-behind) is a legacy segment that should be deprecated

# Message Layouts in SAS FM

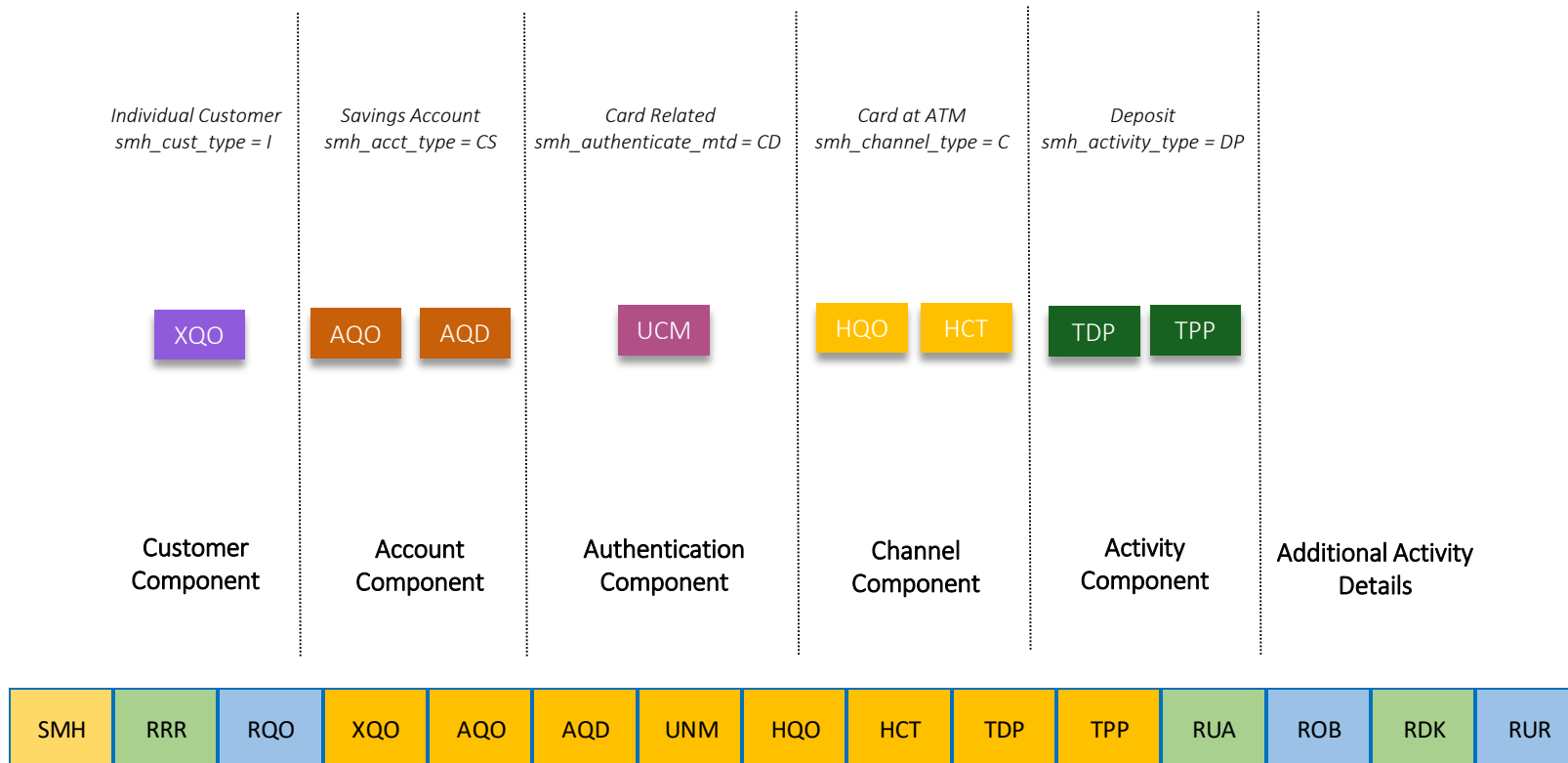
## Example 1: Credit Card Authorization via Chip – on – Chip



## Example 2: Fund Transfer via Mobile Banking with Device Details

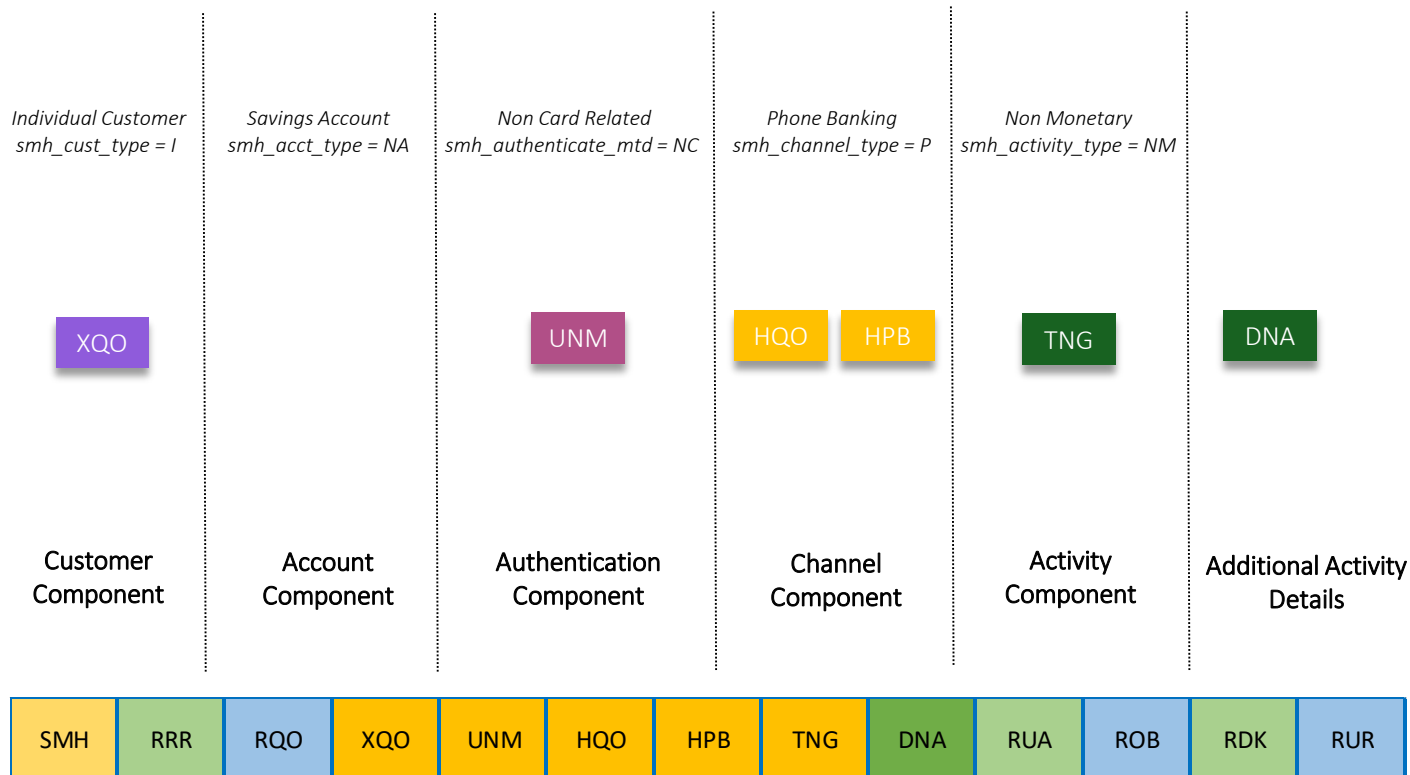
# Message Layouts in SAS FM

## Example 3: Check Deposit at a Non Chip ATM Using Debit Card



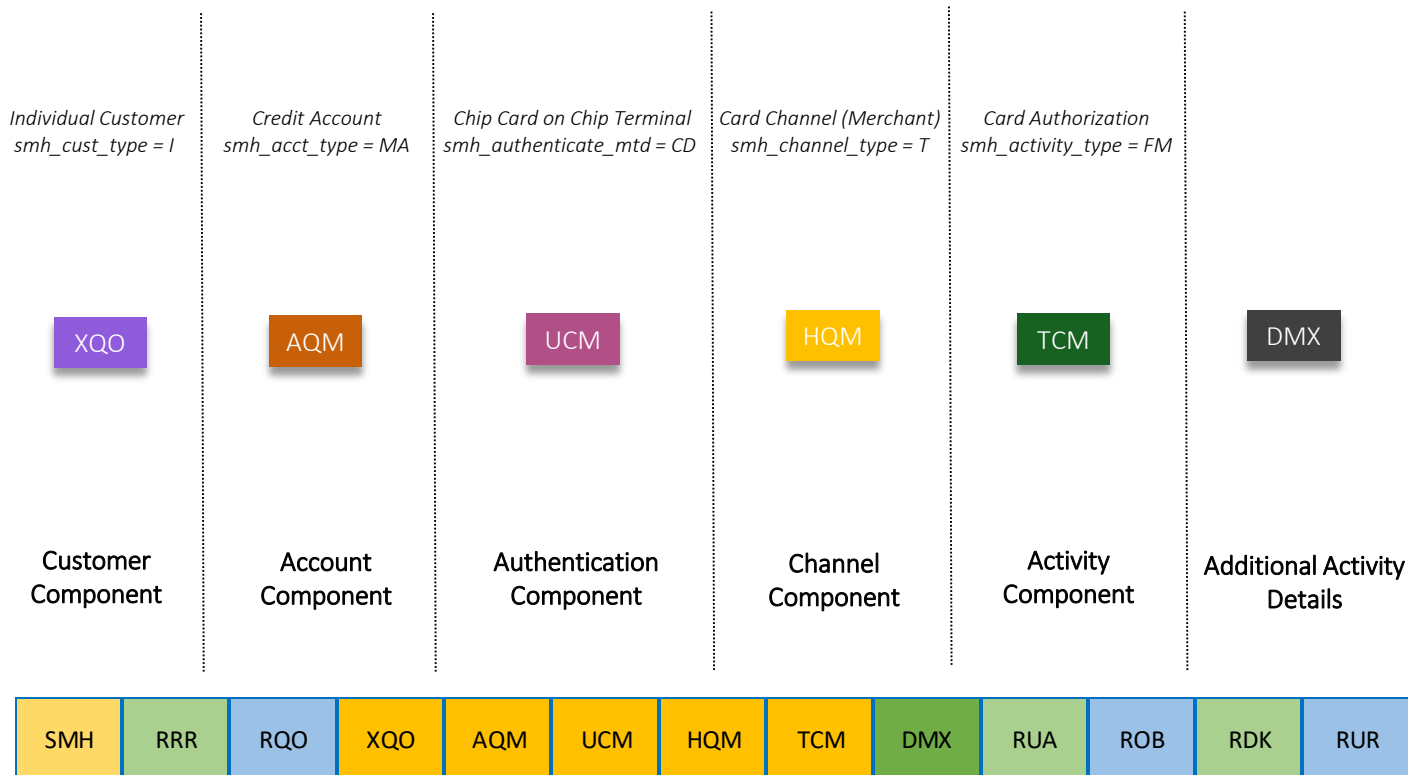
# Message Layouts in SAS FM

## Example 4: Customer Level Address Change via Phone



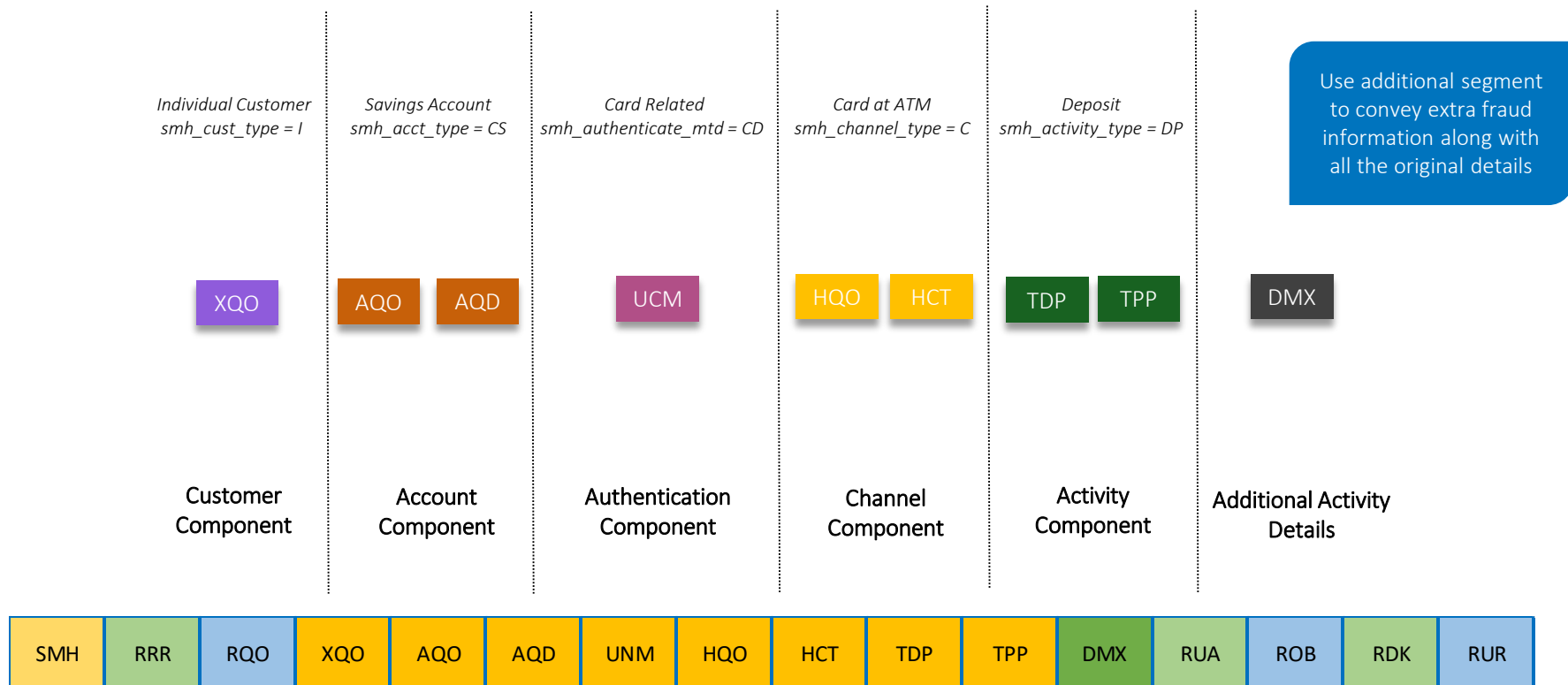
# Message Layouts in SAS FM

## Example 5: Card Authorization on Merchant Account (Non-Chip)



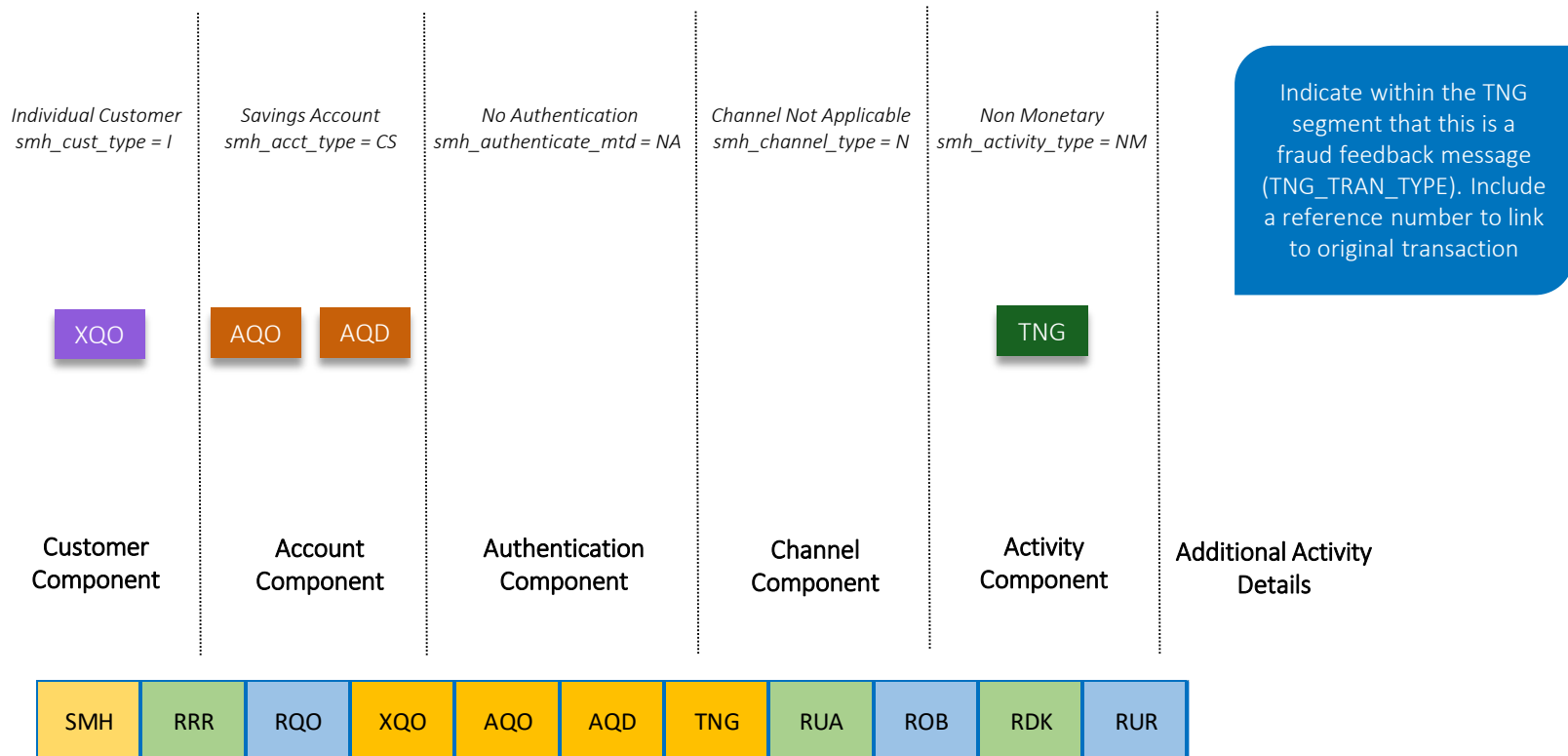
# Message Layouts in SAS FM

## Example 6: Fraud Feedback with Full Transaction Details (e.g. 3)



# Message Layouts in SAS FM

## Example 7: Fraud Feedback as a Non-Monetary (e.g. 3)





# Message Layouts in SAS FM

## An 'Unofficial' Tool for Layout Validation



# Layout Validator



# Message Layouts in SAS FM

How to Read the API Document

# Message Layouts in SAS FM

## How to Read the API Document

- Versioned using a 3 digit sequence
  - E.g. 04.04.04
- The most important work sheets are:
  - **Transaction definitions**: contains a matrix of segments associated for different values of the key SMH fields
  - **Segment definitions**: contains the details of all fields grouped by segment
  - **Allowable**: a list of all possible allowable segment combinations (every possible combination is not possible due to improper context)
  - **NonMon Trx Type**: a list of different non monetary activities specifiable within the TNG segment. New values can be added by the users.



# Message Layouts in SAS FM

## Review of Common Segments

# Message Layouts in SAS FM

## Some Remarks on Segments

- Every segment starts with a *seg\_id\_version* field
  - Serves as a versioning system for individual segments
  - Exact value can be found in the specification document
  - In theory only this field needs to be populated to be accepted by the ODE; all the others can be populated as blanks.
- Field order within segments is fixed; any offsets will produce errors.
- Various entities have two corresponding random digit fields
  - E.g *hgo\_card\_num* (card number) has *hgo\_sas\_card\_rand\_dig* and *hgo\_card\_rand\_dig*
  - Every entity is assigned two different, but fixed random digits via these fields
  - The ‘SAS’ version is automatically generated by SAS FM
  - The other version is automatically generated by SAS FM if it is not explicitly generated by the orchestration layer
  - Useful for sampling at a given entity level

# Message Layouts in SAS FM

## Review of Common Segments



### Field Level Details



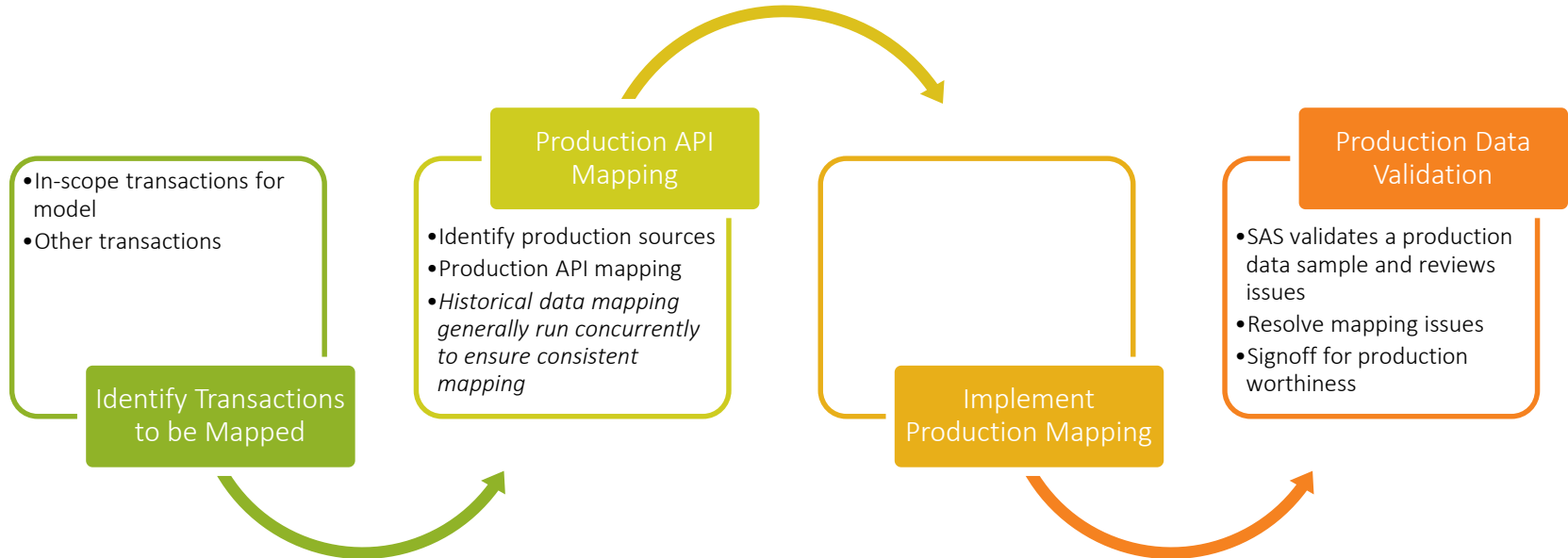
# Message Layouts in SAS FM

## API Mapping

# Message Layouts in SAS FM

## API Mapping Process

- API mapping is the exercise of mapping information from the customers' source systems into the SAS message layout
- Suggested process:





# Message Layouts in SAS FM

## API Mapping

- Should populate as many standard fields as possible
- Also should elicit as much other information as possible and map it into general purpose segments
  - Good domain knowledge about fraud and good intuition on the value of different fields for a building an effective fraud model will be immensely useful
  - Hence the model developer should be fully engaged in this exercise
- Should thoroughly validate samples to ensure that the fields used by the model are mapped consistently with how the model expects them
  - Inconsistencies can completely derail model performance
  - Is generally not an issue if building the model from consortium data



# Message Layouts in SAS FM

TDR and Consortium

# Message Layouts in SAS FM

## TDR

- TDR is the database in which all transactions that flow through the system are eventually stored.
- TDR contains one table for every smh\_acct\_type and smh\_activity\_type combination.
  - E.g. CSCA / CCCA / NANM / SLBF
  - A superset of all segments corresponding to all smh\_cust\_type, smh\_authenticate\_mtd, smh\_channel\_type and smh\_activity\_details values are part of these tables
    - Makes them very wide tables
  - Various additional segments that contain post processing (rule firings, analyst markings) and DB information (last updated, unique transaction ID)

# Message Layouts in SAS FM

## TDR

- Refer to the **Usage Definitions** worksheet in the API document for table layouts
- E.g.

Mid-Tier Base Table Name																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																	
	TDR	Column	Count																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																														
	Transaction	Acronym																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																															
Account	Component	Value																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																															
Activity	Component	Value																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																															
			Description																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																														
CCBF	945	CCBF	CC	BF																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																													</

# Message Layouts in SAS FM

## Consortium Process (9000 Jobs)

- Most of SAS FM customers send a nightly or weekly dump of all transactions from the TDR
  - With all PII information obfuscated properly
  - May contain analyst markings which can be a source for fraud transactions
- One file per TDR table
  - Though not all files may be sent depending on scope
- Sent as gzipped files
  - Special utilities required to read these files
- This data is used for various purposes:
  - Model rebuilds
  - Production analysis and troubleshooting
  - If permissions exist contractually, then can be used for building consortium models or supplementing data during other model builds

# Message Layouts in SAS FM

## Fraud Tagging (5006 Job)

- Primary objective of fraud tagging is to combine and reconcile fraud data from different sources to tag transactions in the TDR
  - Most customers have sources outside SAS FM where complete fraud reports are maintained
- Primary input is a customer provided file containing fraud transactions
  - Has a separate message layout specification (fraud API)
- Tagging 'matches' these transactions to transactions in the TDR
- Establishes a mechanism to send periodic fraud consortium to SAS for future model rebuilds
  - The input fraud file itself is sent back to SAS (fraud consortium)
  - Tags are written out to the TDR which also will be available via consortium feeds
- Various production uses for fraud tagging:
  - To produce performance reports (analytic reports)
  - Rule-estimation
  - CPP