TOBACCO CONTROL
ON TRACEABILITY AND SECURITY FEATURES

Review of IT system architecture and IT system specification after Interim Report III – Second draft

Version	Date	Author	Notes
1.0	24.05.2017	[redacted]	The first version

[redacted]
mobile: [redacted]

Contents

mobile: [redacted]

1. Summary

1.1 Context

This document was written as a reaction to and based on Interim Report III – Second draft document thereinafter as Interim report or IR for **Implementation analysis regarding the technical specifications and other key elements for traceability and security features in the field of tobacco products** and after workshop with Experts 17th May 23, 2017, in Brussel.

All the statements placed in this document are based on

1) facts presented in Interim report if not obvious with chapter identification or

2) facts presented as standards and standard patterns also with the identification of the source or standard itself.

1.2 Objective

The aim of this document is to summarize questions, discrepancies and other thoughts in a counter proposal manner, to point to the parts of the current solution design that should be reconsidered.

2. Main architectonical components

The aim of this chapter is to analyze the purpose, characteristics and the value of the three main entities which the system is designed to be built from.

2.1 Solution design summary

The Interim report defines three main architectonical parts of the solution – ID Issuer, Primary data storage and Surveillance data storage. These three entities are designed to be mutually independent in case of development, operation and also communication with the data consumer which should be realized via messages.

Here below, the statements about these three main entities are summarized. The aim of these statements is to bring a picture of these three entities purpose, its relation and characteristics important for further thoughts. The aim is not to describe these entities fully and comprehensively.

There is missing information about data mining and deep data analysis. Request for adhoc reports, data analysis is needed standalone copy of PDS, SDS for preparing any kind of reporting without any impact to systems performance, system availability (see Ch. 2.4. for more detail)

[redacted]
mobile: [redacted]

2.1.1 Common characteristics of both types of data storage

- Data storages in general (IR Ch. 4.5.5.1 and IR Ch. 4.5.5.2)
  - Handle message in two steps
    - Data acquisition
      - Authenticate the sender
      - Verify the compliance with expected schema
      - Store the message as is
      - Return back to the sender an acknowledgement positive or negative according to previous step results
    - Data processing
      - Send a copy of message to SDS (PDS only)
      - Clean data (filling missing values, smoothing noise…)
      - Check completeness, consistency, accuracy and reliability of data
      - Consolidate and store the message data
  - Two types of access – two types of interfaces
    - Data exploitation such as reports, data analytics, bulk extraction or alert tools over successfully consolidated data
    - Standard and secure interface for data exchange (in/out) with authorized external systems
  - Other consideration
    - Horizontal scalability for both storage and processing capacities
    - Data back up as a mirror regardless of redundancy level used and in near real time
    - Tiered data storage for better data consumption and analytics performance
    - Inactive data automatically archived with possibility to restore

2.1.2 Entity specific statements

- Primary data storage (IR Ch. 4.5.5.1) thereinafter PDS
  - One instance per manufacturer/importer or group of them
  - Receives messages from
    - manufacturer's/importer's Temporary buffer
    - distributor/wholesaler or IDI routed thru Repository router
  - Handle these messages the common way described before
    - As a part of message handling sends a copy of all received messages to SDS
- Surveillance data storage (IR Ch. 4.5.5.2) thereinafter SDS
  - One centralized for whole solution
  - Includes Repository router thereinafter RR
    - Receive all messages from IDI and distributors/wholesalers
    - Route the messages to appropriate PDS according to the data
  - Receives a copy of all PDS messages and handle it the common way described before
- ID Issuer (IR Ch. 4.5.5.3) thereinafter IDI

mobile:

- One centralized for whole solution
- Generates unique serial numbers on request from manufacturer/importer or distributor/wholesaler thereinafter UIDs
- Sends generated UIDs to requestor and SDS as a prior

2.1.3 Inter entity communication commons

Three main entities described above are designed to communicate to each other via messages. This message handling was described as PDS and SDS common characteristic but it's partially also valid for IDI.

- Message handling common part for all entities are at least: (IR Ch. 4.6)
  - Data acquisition
    - Return back to the sender an acknowledgement positive or negative according to previous step results thereinafter ACK
  - Data processing

2.2 Question of three entities design

The first question should be if these three main entity designs are the proper ones and why. It's obvious that the volume of processing data, in the sense of message count, is a good reason to put a huge accentuation on the distributed message handling. Also, the need of centralized data reporting or redistribution, for example, per member state, is evident and has to be provided somehow.

The proposed solution covers these needs but the question is if it is in the most efficient way in terms of:

- Infrastructure load during inter entity communication
- Implementing, operating and executing the same process multiple times

2.2.1 IDI and SDS relation and communication

Both IDI and SDS entity are designed to be each of them, one centralized instance for the whole system. For both these entities it does make sense.

There are well known solutions based on distributed principles to provide unique IDs as, for example, public IP address distribution system. These systems have their disadvantages connected with the ID range dividing and distribution. So, if the one IDI instance solution is efficient enough to handle all UID requests then it could be the correct choice.

All UID requests are sent directly to the only one IDI by all manufacturers/importers and all distributors/wholesalers (IR Ch. 4.5.5.3). All generated UIDs are, before sending back to the requestor, reported to the SDS and afterwards via SDS RR and also to the relevant PDS (IR Ch. 4.6.1.1).

mobile: [redacted]

There is missing information if the SDS or PDS stores the information which UIDs have been provisioned in this step. According to the IR Ch. 4.2.2.2 the information should be stored in SDS but not transferred to the PDS.

- ✓ So the information in Ch. 4.2.2.2 and Ch. 4.5.5.3 is in conflict with each other or at least incomplete. Based on the presumption that the document isn't in conflict with itself, the information in single chapters is just incomplete and the UIDs reported to the SDS are stored here in this step and also routed to the relevant PDS and stored there too.

There is also missing explicit information if the IDI has to wait for SDS ACK response or not in the IR.

- ✓ Nevertheless according to Request for serial numbers diagrams (IR Ch. 4.6.1) and IDI responsibility to *Notify SDS solution of which UIDs have been provisioned. This notification shall be done prior to delivering UIDs* (IR Ch. 4.5.5.3 / b) it's possible to assume that the mentioned SDS ACK has to be received before delivering UIDs to the requestor. Otherwise, there is no guarantee that the information was delivered to the SDS at all.

In that case, the requestor doesn't get any UID without or before the SDS is informed about that.

- › **When there is only one IDI and one SDS and all IDI generated UIDs are reported to SDS before it's sent to the requestor why couldn't it be generated by SDS or by IDI as a part of SDS as the Repository router is? IR proposed approach doesn't avoid SDS response availability dependency and it brings the necessities of message exchanges and processing between these two entities, probably over WAN.**

2.2.2 PDS and SDS relation and communication

The idea of separated PDS per manufacturer/importer thereinafter M/I or group of them as a method to balance a workload of requests for data storing or validation seems quite good.

New UIDs are delivered to the M/I thru RR directly from the IDI. So the M/I needs to communicate with SDS just to store the information about:

- the UID on the product placing;
- parent child linking during products aggregation;
- and product shipping.

Also the requests from distributors/wholesalers thereinafter D/W are routed to the relevant PDS. It means to the PDS where the UID was registered originally. This communication covers:

- the UID validation;
- damaged or other reason unreadable UID handling:
  - new UID on the product placing;
  - parent child linking during products aggregation;
- and product shipping.

All this information needs to be available across the PDSs at the end to reach the requirement of overall picture reporting and querying. With SDS designed as an centralized data storage entity, it's a natural way to collect the information from all the PDSs in the SDS.

mobile: [redacted]

The current solution design is set up to transfer the information to the SDS via resending original messages after its validation. It means that messages received by SDS from PDS will be very likely valid, but the process for message handling will be fully executed a second time for the same message.

- **When the data should be cleaned and consolidated at the end in both PDS and SDS in the same way, why it couldn't be done only once? The data could be transferred to the second storage, already transformed and probably in a more efficient way based on:**
  - **higher level of data consolidation / normalization;**
  - **and possibly a larger data volume in a single batch.**

SDS as is designed, has to be able to process all the messages processed in all the PDSs together for the same long term period. It means, for example, in one day SDS has to process all the messages processed by all the PDSs during such a day. Otherwise, if there were any unprocessed messages left from the first day, there is hardly a possibility to process them the next day with all the messages from this next day.
If we admit that the manufacturer production line operates 24 hours a day, then there is no time to catch up with the delay after work and SDS has to process all the messages near real time like PDSs do.

- **When the SDS has to be able to process all the messages for all PDSs at close to the same time, why couldn't these messages be processed only in the SDS? In that case, the messages would be processed only once all together and the data stored in only one centralized storage with the ability to provide the required overall reporting.**

2.2.3 Inter entity communication discrepancies

To summarize the rules in the message forwarding specification in the IR document and mentioned before:

- SDS RR is specified to route all messages from IDI and D/W to relevant PDS (IR Ch. 4.5.5.2)
- PDS sends a copy of all messages to SDS (IR Ch. 4.5.5.1)

Information from IDI is reported and stored directly to SDS (IR Ch. 4.2.2.2) and routed later to the PDS (IR Ch. 4.5.5.3)

- **When the IDI information is stored to SDS directly and also reported to PDS where it is needed, then forwarding this message by PDS to SDS is redundant and therefore an unnecessary infrastructure load.**

Figure in IR Ch. 4.6.2.2 shows that event reporting messages sent by D/W to SDS should be processed and stored directly and afterwards resent to PDS.

- **When the D/W reporting information is stored in SDS directly and also reported to PDS where it is needed then forwarding this message by PDS to SDS is again redundant and therefore an unnecessary infrastructure load.**

2.3 Data discrepancy notification

The solution is designed to identify and handle discrepancies in data processing as, for example, a request for validation of already deactivated UID is possible.

mobile:

This identification and handling in the sense of reporting to the competent authorities is described for the first time in Process diagrams, located in IR Ch. 4.2.2, as the responsibility of SDS.

The definition of event reporting for M/I (IR Ch. 4.6.2.1) specifies that the discrepancy is identified by PDS and reported to the SDS, which reports it to the competent authorities. SDS sends ACK to PDS for the discrepancy report message reception. SDS also sends a positive ACK to PDS if it doesn't identify any discrepancy during the validation process, where PDS hasn't identified any as well.

The document doesn't specify what the steps are for the competent authorities and what is expected to be done with the defined notification. There is also no discrepancy feedback to the M/I or D/W as a primary UID operator.

- **When there is a system of UID validation and discrepancy identification in the system designed, why are the results of this kind of UID check not reported to the primary UID operators? This has to lead to continued use and distribution of invalid UIDs and also the repeating of unsuccessful validations and its storing and reporting. If not, the process is not described in the document.**

2.4 Data storage capacity and differentiation

2.4.1 Data storage capacity

The system is designed to back up all the data stored near real time on both data storage levels and independently on primary data redundancy (IR Ch. 4.5.5.1 and IR Ch. 4.5.5.2).

This requirement brings an assurance of immediate data availability in the case of the PDS/SDS entity system primary data storage or data transformation failure. It also brings a doubled requirements of data storage capacity.

The document specifies the system space consumption per year for PDS and SDS separately and minimal data retention period of 10 years for both data storage levels.

The document doesn't specify data availability and backup requirements during this minimal period.

- ✓ Then it's assumable that the data should be available and backed up under the same conditions for all the minimal period.

Usual period between tobacco product manufacturing / importing and the last economic operator before the first retail outlet is assumed to be about 3 months. Nevertheless the system is also designed for audit reasons and it's understandable that the period required for this purpose can be much more longer.

On the other hand reporting and audit solutions not described in details in the document usually has much lower requirements to data availability latency then operation data layers like SDS/PDS in communication with M/I and D/W is.

- **When there is differentiated usage of data stored in the system during minimal time period required why isn't this differentiation a part of data storage requirements specification? Current specification requires the availability and durability of primary database system for**

mobile:

all the time period when for the main part of this period the data stored will be used for reporting and audit purpose only.

2.4.2 Data storage purposes and tiers

The previous chapter has opened the question of differentiated data storage purposes. There is a requirement of tiered storage mentioned in IR document (IR Ch. 4.5.5.1; IR Ch. 4.5.5.2) and another one for a hosting capability to exploit data such as reporting, data analytics, querying, bulk extraction etc. (the same chapters).

The system is mentioned to store messaging history for the audit trail. But the primary usage should be a storage of UIDs and referenced information for the purpose of validation and status actualization. Therefore the data from messages needs to be consolidated and transformed to the form much more effective for such manipulation. The next and even last but very general purpose specified in requirements and mentioned before should be reporting and analytics which are not specified pretty well.

These differentiated purposes correspond with standards of layered data storage architecture in general despite of that there is no specification which architecture has to be used in the IR document. Therefore in general:

- ✓ Staging area dedicated for raw data received can be historicized.
- ✓ Consolidated layer can be built according the standards to store effectively the full information received.
- ✓ Access layer should be designed based on accessor needs.

The IR document doesn't specify the requirements for reporting or dashboards content. It doesn't define either accessors or their needs.

- ✓ So the data storage can be built as tiered architecture as is required but without the access layer unless its content or detailed purpose will be defined.
- › **When there are some particular purposes or needs for data reporting or dashboards known why aren't these analyzed and specified in the IR document?**

What was mentioned before and is in contradiction with data processing standards and efficiency is a double processing of raw data on different levels of data storage solution.

- › **When there is required tiered architecture of single data storage why aren't the features of tiered data processing used thru the solution in the sense of the only one transformation of raw data to build consolidated information layer as a base for sharing such information.**

3. UID lifecycle processes

The aim of this chapter is to analyze the UID lifecycle as single steps and also referenced processes. The main purpose should be to validate its feasibility and efficiency in the way it was designed.

mobile: [redacted]

3.1 Solution design summary

As the UID is an integral part of all the solution designs, it's mentioned in almost every part of the Interim report document. Therefore, the set of statements summarizing the UID processing is listed here below, as a big picture from the very beginning of this chapter.

- UID generation (IR Ch. 5.1.4; IR Ch. 5.2.4)
  - Initiated by
    - M/I for the first product packing (IR Ch. 4.2.2.2 / 1.1)
    - M/I for the first product aggregation packaging (IR Ch. 4.2.2.3 / 4.1)
    - D/W for re-aggregation packaging (IR Ch. 4.2.2.5 / 10.1)
  - Provided by IDI (IR Ch. 4.5.5.3)
  - Reported to and stored in SDS and PDS through RR (IR Ch. 4.6.1)
- UID application (from printing to verification and reporting)
  - Provided by
    - M/I for the first product packing (IR Ch. 4.2.2.2 / 2; 3)
    - M/I for the first product aggregation packaging (IR Ch. 4.2.2.3 / 5; 6)
    - D/W for re-aggregation packaging (IR Ch. 4.2.2.5 / 10.3; 10.4)
  - Reported to and stored in PDS (thru RR for D/W) with resending to SDS
- UID linking of parent child relation during aggregation
  - Provided by
    - M/I for the first product aggregation (IR Ch. 4.2.2.3 / 6.1.3)
    - D/W for re-aggregation (not mentioned in business process diagrams)
  - Reported to and stored in PDS (thru RR for D/W) with resending to SDS (IR Ch. 5.2.5)
- UID unlinking (IR Ch. Ch. 5.2.5)
  - Provided by
    - M/I for the dis-aggregation before dispatch
    - D/W for dis-aggregation during distribution
  - Reported to and stored in PDS (thru RR for D/W) with resending to SDS (IR Ch. 5.2.5)
- UID deactivation (IR Ch. 4.2.2.6; IR Ch. 5.1.6; IR Ch. 5.2.5.3)
  - Initiated by
    - M/I or D/W based on unreadable status of UID carrier
    - D/W based on any reason re-aggregation (disaggregated aggregation UID only)
  - Provided by PDS (thru RR for D/W) and afterwards SDS (resent message) based on
    - UID read in deactivation mode
    - Number of units to be deactivated (IR Ch. 4.2.2.6)

mobile: [redacted]

3.2 UID deactivation

3.2.1 Identification of UID to be deactivated

The process of UID deactivation, as was specified, can be triggered based on two main reasons 1) re-aggregation process occurs for any reason different from the second reason or 2) the UID is unreadable with what can appear in any validation step during the whole process.

The process of deactivation begins with the step of UID scan in deactivation mode independently on the deactivation reason mentioned before.

The IR document doesn't specify what the deactivation mode is. If for example, the reader equipment switched to this mode has a better ability to recognize the UID at the cost of bigger time consumption or just triggers the deactivation process instead of validation.

- ✓ If the document doesn't specify such a mode as an equipment property requirement; the meaning of the mode can be just a method how to influence which process validation or deactivation has to be triggered.
- › **When there is no possibility to switch the UID scanner to another mode from the perspective of UID recognition ability, why does the process of deactivation remain unreadable while the UID begins with reading the UID?**

When the UID is still unreadable the second step for its deactivation is to *disaggregate to an unnecessary level* and then based on the readability of inner UIDs, *Report the UIDs to be deactivated* or *Report the number of packets to be deactivated*.

The IR document specifies that readable UIDs in a package with unreadable UIDs should be deactivated too instead of unlinking and parent UID deactivation. This situation is better specified in IR Ch. 5.2.5.2 as a part of the disaggregation process, which covers aggregation UID deactivation based on inner UIDs and parent child linking is what seems much more reasonable.

- ✓ Again if the document defines the same process twice in not precisely the same way let's assume that the definition is just incomplete or inaccurate and the solution should be a union of these definitions.

The second case when there is no next level for the disaggregation of a packet with an unreadable UID is designed to be handled by PDS/SDS based on the number of unreadable packets.

The IR document doesn't specify how the PDS/SDS should identify UIDs to be deactivated based on its count. This identification could be done, for example, based on all UIDs dispatched on one side and received on the other side. But this example is based on the synchronized transmission when all the units from one delivery should be received before another one is dispatched, is what probably doesn't correspond with reality.

- › **Then there is missing specifications how to identify unreadable basic, not aggregated units UIDs during the deactivation process in the IR document.**

mobile: [redacted]

3.2.2 Deactivation process executor

The deactivation process has to be initiated by the M/I or D/W based on unreadability of processing UID or any other need for package re-aggregation. During this process up to three steps has to be executed on the data storage level:

- Identification of UID to deactivate (if unreadable)
- Unlinking of parent-child relation (if UID involved in aggregation)
- UID status change to deactivated

In other words the data storage entity is the part of system that is responsible for the UID to be deactivated identification, in the case of unreadable UID and therefore the deactivation message with UIDs count only. The output of this identification process is a list of UIDs to deactivate, more precisely it should be the message with UIDs to deactivate.

Figures in IR Ch. 4.2.2.6 shows that the deactivation process executor data storage differentiates according to the initiator. M/I processes the deactivation in conjunction with PDS and D/W with SDS.

On the other hand, based on the messaging system architecture the SDS in this figure could mean just RR module to resend the message to the PDS. If so, according to the messaging system rules such a message should be resent specifically to the SDS and processed there as well.

› **When the UID identification has to be implemented and executed on PDS where all the messages should be routed to, why is this message initiating UID identification processed also in SDS, instead of identified UIDs to deactivate results forwarding only? If the process is meant to be designed this way this, it is not specified in the document at all.**

4. Change management and maintenance support

Change management must include consistent implementation of change implementation across all systems. All changes must be approved by an architect.

These changes must be implementable in all systems at the same time (in case of impact). These changes need to be enforced even if the systems are managed by multiple providers.

What is missing: There is no definition of non-production environment (DEV, CAT, UAT), data. Establish an independent IT commission that will approve IT architectonics for the changes. The build and provide of more environments has an impact on project costs (OPEX and CAPEX).

mobile:

5. Inter communication specification (messages, interface)

Detailed status descriptions need to be added to states if the required workflow is not running, did not work, unexpected was suspended. It is also necessary to define the expected quantities, ranges, etc.

What is missing: list of errors messages for communications between interfaces (or aplications).

List of status, status check if they are in correct order (steps):

Platform error message	Error description
Invalid or missing integration API Key	The integration API key is either incorrect or has not been included in the API call.
Integration is not active	Your integration is not active. Try to publish your integration first.
Account is not active	The account is not active.
Empty message content	Message content is empty.
Empty receivers list	Destination list is empty.
Message receivers list is too long	Too many destination addresses.
Two-way integration error – You should specify correct from number	Sender number was not specified or is not connected to your integration.
Two-way integration error – You should use two-way integration to specify from number	User specified from number but integration isn't two-way.
Unsupported charset. - [charset] is not supported	The specified character set is not supported.
ScheduledDeliveryTime format is incorrect	ScheduledDeliveryTime format must be yyyy-MM-dd'T'HH:mm:ssZ
IP lock down violation. – Requests from [ip] forbidden according to settings	Couldn't determine user's IP address or it is not in whitelist.
Maximum message parts exceeded. – PartCount is: [partCount]	The text component of the message is greater than the permitted 160 characters (70 Unicode characters).
Message exceeds max available characters: [defaultMaxMessageLength] – Please enable message parts to send the message	The text component of the message is greater than the permitted 160 characters (70 Unicode characters).
Duplicated destination address found: []	The destination number you are attempting to send to is duplicated.
Number opted out	The user has opted out and is no longer subscribed to your service.
Invalid destination address	The destination number you are attempting to send to is invalid

mobile:

[redacted]

mobile: [redacted]