TOTSCO

TOTSCo Hub Specifications & Technical Requirements

Technical Design

Version Draft V0.3

TOTSCO

Contents

TOTSCO

TOTSCO

TOTSCO

Figures

TOTSCO

1. Introduction

The One Touch Switch Technical Design document complements and supports the One Touch Switch Process Industry document, v4.0 issued 18th August 2022. This document provides the technical definition and requirements for all software deliverables forming the TOTSCo Hub and supporting the switching processes. This document will progress through several iterations as engagement with vendors advances from the RFP process onwards and as specifications of the messaging protocols are ratified and approved by the communications industry consuming the One Touch Switch process.

For a definition of the One Touch Switch process, please refer to the One Touch Switch Industry Process document referred to above.

1.1 Document release roadmap

The following list of targeted release versions of this document provides expectations of how this document will evolve, its purpose, and its content.

Target Release Date Status	Reason for change
v1.0 Sept 2022 Draft	First release to support the RFP process by documenting all currently known and discussed IT requirements, and defining the messaging structures and contents to allow CPs to begin design and development activities. The first proposal for business switching transactions is also included for reference.
v2.0 T.B.D. Not started	Updates to further detail the IT solutions, providing detailed functionality and workflows. Further refinement after consultation on messaging structures and alignment with industry process design where needed. Goal to become an input document to the selected vendors' solution design process.
V3.0 T.B.D. Not Started	Final specification agreed with vendors and CPs for production requirements and messaging specifications.

TOTSCO

1.2 Change Log

Version Date Changed By	Reason for change
V0.1 First draft 4/9/2022 OTS-DDG	First draft output from the OTS-DDG (design drafting group) on behalf of TOTSCo. Intended for CPs only as a technology definition, specifically providing message formats for the OTS process. Only issued for review within the DDG.
V0.2 Updated draft 19/9/2022 TOTSCo	Draft to define the scope and technical definition of required deliverables to meet and support the One Touch Solution Process.
V0.3 Draft for release 30/9/2022 TOTSCo	Clarrification and technical implementation updates

1.1 Material changes since release 0.3

Corrected use of dictionary and stadardised on directory.
Added new error messages to the postoffice API.
Added final URL formats for developers to build against.

1.2 Contributing Authors

Author	Organisation
Dave Stubbs	Virgin Media
Niall Gillespie	BT
Nick Holland	8x8

2. Scope

The OFCOM regulations in C7 of the General Conditions of Entitlement identify the terms of reference to protect domestic and small business customers when changing communications providers, relocating, or changing service.

The switching process is limited explicitly to IAS and NBICS for this document and the associated CP community, although C7 also refers to mobile service as well, which is not within the scope of the TOTSCo organisation.

Through an extensive industry consultation process, starting in 2019, the industry has reached an agreement on a switching process acceptable to all CPs providing a consumer switching process and is documented in the One Touch Switch Industry Process document.

OFCOM has been clear with the industry that a GPL switching process must also be provided for business customers of all sizes, again limited to IAS and NBICS services.

As of this version of the document, the requirements for business are still being defined, but the currently proposed solutions are included within this document for completeness, and to inform any potential vendor of the anticipated scope of a full technical solution.

3. What is the TOTSCo Hub

When the industry was discussing how a switching process should work, one of the core principles of all proposed solutions was that all providers required a centralised hub where all message interchange occurred. While the term hub has stuck, the scope of the solution has expanded to include other technical components required to deliver an OTS and GPL experience. This document will define all aspects of the required technical solution from a functional perspective. How all the requirements are met will be determined in conjunction with the selected vendor(s) as the solutions are seen as being composed of multiple core functional elements that could be delivered separately if so desired.

3.1 A Messaging Service

At the very heart of the OTS and GPL processes is the need to exchange messages between two parties, known in switching as the gaining and losing provider. The following core principles were agreed upon within the DDG in defining the high-level function and capability of the messaging service with the best industry parallel being that of a postal service. Therefore, the concepts of a post office and letterbox will be used generically within this document to define these entities and avoid confusion with the term hub.

The messaging service will provide a synchronous means of delivering messages to the central post office. The post office exposes a letterbox, a web service in effect, allowing messages to be posted to it and to acknowledge receipt, much like a registered letter.

The post office is allowed to look at the message to determine who to send it to and what the message type is. That allows the post office to look up its directory to identify where their letterbox is. This will be another recipient-hosted web service.

The following very simple diagram shows the basic application of this messaging delivery process.

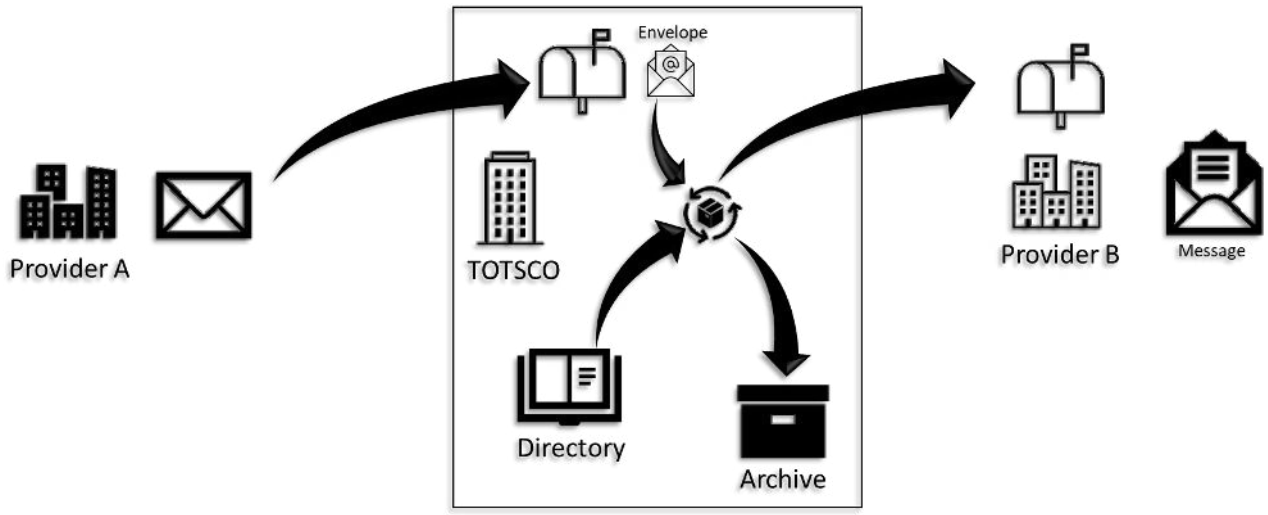

Figure 1 – Messaging Service Basic Concepts

In the diagram, provider A posts the message to the TOTSCo post office letterbox. The post office then inspects the envelope to determine the recipient's information. The directory is inspected to determine where that message needs to be sent and then the post office delivers the message to provider B at that specified letterbox location.

The post office also forwards a copy of the message to an archive. From here any reporting, regulatory processing, financial accounting etc. can be performed.

The post office does not need to understand the message contents, only the envelope in which the message is contained.

A detailed specification of the messaging service is defined in greater detail through this document.

3.2 Post Office Administration/Operational Portal

The post office must provide an administration portal to allow monitoring, configuration, and troubleshooting of the post office operations. This can be combined with other portals where there are obvious benefits in doing so.

Monitoring would include the state of RCPID letter boxes, whether they are receiving messages or not, transaction volumes and so on.

Detailed message breakdowns will be performed by the archive. The hub monitoring is purely for operational status information and not for breaking down transaction volumes.

3.3 The Directory and Administration Portal

The directory portal is where both TOTSCo and the retailers maintain the configuration information related to the supported retailer and the letterbox configuration information the post office needs to perform its function.

TOTSCO

3.4 The Archive

The archive is a persistent data store of all messages being processed through the post office. The information is held for a period before being purged. Storage policies may differ depending on the message type but GDPR must be adhered to.

The archive should only hold the data for as long as it is required, the exact content and durations are to be defined once the operational and reporting requirements for OFCOM and the industry by TOTSCo are defined.

3.5 The Reporting Portal

A reporting tool will be provided, accessible via a portal to define and deliver standard reports to TOTSCo, OFCOM and retailers. Retailers must only be able to report on the messages sent to or from them. Most reports will be summary/statistical, though reporting of a specific message interchange for a single switching transaction should be supported.

It is expected the reports will be created in CSV format with ranges of selectable values, such as date and time as well as filters, for example on the SOR or a specific RCPID.

3.6 [Redacted] [Redacted]

3.7 [Redacted] [Redacted]

3.8 [Redacted] [Redacted]

TOTSCO

3.9 Accounting System

The TOTSCo hub is operated on behalf of the industry and will require that an accounting and invoicing function is provided to automate the process of counting chargeable events and producing the appropriate invoices.

A store of invoiceable entities will be required, mapping the transaction identifies from the directory to those parties so invoices can be aggregated where necessary over multiple messaging processes or parties.

The calculation mechanism for invoices is yet to be defined (September 2022) and will be provided at a later date.

The invoicing function will require its reporting, data storage and archival functions separate from the transactional data stores.

4. Post Office Operating Models

Before looking at the details of the solution architecture, some background on the core assumptions and premises for the operation of the post office solution is required to put the following sections in context and to define some generic terms without assumption or prejudice towards any specific party or role.

Initial design assumptions for the post office aligned specifically to the OFCOM regulations of a provider-led switching model where retailers are obliged to provide a switching service for customers.

In its very simplest term, that would give rise to a simple retailer-to-retailer model shown below.

Figure 2 – Retailer to Retailer Model

Now in the consumer switching model, several large retailers will wish to interact directly with the post office, though there may be instances where some smaller retailers have their services provided for them and managed by a larger wholesale entity. In those cases, the wholesale entity may provide the integration to the post office on behalf of the retailer as shown below.

Figure 3 – Retailer via Wholesaler Model

In this model, the wholesaler is providing all of the service integration to the post office on behalf of one, or maybe many retailers, and therefore can represent a single common connection point to the post office for all of those retailers' messages.

The same is true of third-party integrators that could deliver a service capability to the post office on behalf of customers, and the post office itself is unaware of the upstream implementation architecture, only the origin and destination of messages.

Figure 4 – Retailer via TPI Model

A further model for the post office is where an aggregation function is provided for messaging on behalf of many retailers. [Redacted]

Again, to the post office, this should simply look like a wholesaler or a 3rd party integrator where the same locations are used for the receipt and delivery of messages.

[Redacted]

These principles help to promote a flexible deployment model for the switching processes by allowing the delegation of message generation or delivery responsibility to any party or system on behalf of the retailer.

This approach allows for business switching to be adopted through the hub using complex supply chains by allowing delegation of control to any party in a chain on behalf of any given retailer.

If a retailer uses multiple wholesale partners who offer services to perform the switching function on behalf of the retailer, there is a potential for a switch to be split across wholesalers. If this is the case, the retailer must identify a single aggregator for responding to the match transactions and managing the switching activity. [Redacted]

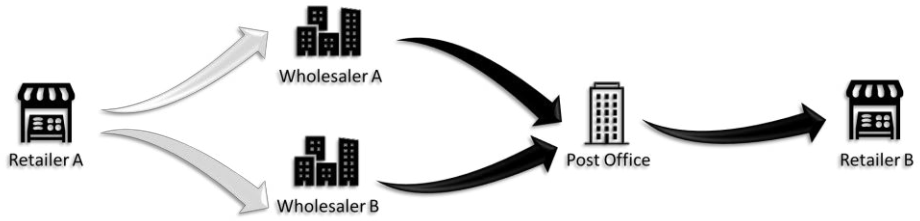

Figure 5 – Retailer via Multiple Wholesaler Model

If a retailer follows a model of a single supplier, always providing services to a customer from one wholesaler or another then the wholesaler can act on behalf of the retailer [Redacted] to perform the aggregation function.

It should be stated here that there is no requirement for wholesalers to be integrated into the Post Office, just that some may choose to provide a simpler integration model to their retail partners.

It may be more practical to use a TPI to handle multiple wholesaler integration to the hub, but that decision would rest with the retailer and wholesalers to ensure they can deliver a compliant switching capability.

Taking the post office requirements a step further to support other messaging processes and entities other than retailers. The message structure for processing by the post office will be open and extendable, allowing other message formats beyond switching to be processed by the post office. Number porting may be an ideal future candidate for moving to the post office as an example, and the messaging in that model would not be between retailers, but between voice CP's (holders of CUPIDs). Therefore, the directory, and the message delivery headers must support a combination of delivery entity type and ID. This is a very simple design decision but opens up the potential use cases for the post office for future adoption of other inter-industry processes.

4.1 Naming conventions

Given the potential for confusion regarding the roles of individual entities described above and their relationship with the post office, the following terms will be used to represent multiple actors and roles.

Term	Definition
Retailer	The specific entity, as known to the customer from their bill as the organisation providing their service and not necessarily the network operator, wholesaler, sales agent etc.
Originator	In a message exchange, the entity in the supply chain who will generate and send messages to the post office. This could be a retailer, a CP, a wholesaler, a TPI or an agent depending on the nature of the supply chain involved.
Recipient	In a message exchange, the entity in the supply chain who will receive messages from the post office. This could be a retailer, a CP, a wholesaler, a TPI or an agent depending on the nature of the supply chain involved.
Delegate	Any party that acts on behalf of a reseller to manage the switching process on their behalf.

5. Detailed Requirements

Where possible, this document will refrain from defining specific technologies or deployment patterns that the engaged vendor must follow. There are however some key exceptions to this that the industry requires a definition of now (September 2022) to begin their design and development activities. Specifically, this refers to the messages delivered to and from the post office where the core principles will be defined in this document and will be subject to immediate change control.

All portals must be WCAG compliant, and all deliverables must be hosted solutions and not require any deployment of the vendors' technologies on retailers' systems.

While the solution is being procured to meet the requirements of the OTS and GPL processes mandated by OFCOM, the architected solution has considerable potential benefits to the communications industry in other initiatives. They are not within the scope of this deliverable, or TOTSCo at this time, but where a design decision is specified in the requirements for a system or process to remain agnostic of the switching process itself, this is because the solution can be seen to have future capabilities that the industry will consider expanding to in future and building in any switching specific design or logic could restrict the future potential of that solution.

6. Post Office

The post office function is by its very nature intended to be process and message neutral. It is intended to not be specifically designed just for OTS and GPL, and all aspects of its design should maintain process neutrality while meeting the core functional requirements of the switching workflow.

The post office has as its core functional responsibility the requirement to allow messages sent to it to be onward delivered to the intended recipient to a location of their choosing. Messages must be delivered by the post office to the recipients' letterbox in the order they are received.

Every message type will have a delivery policy defined within the post office so that it can understand how to handle that message if it cannot be delivered because the recipient's letterbox is not available.

The following diagram lays out the principal processes expected to deliver a resilient post office function.

TOTSCO

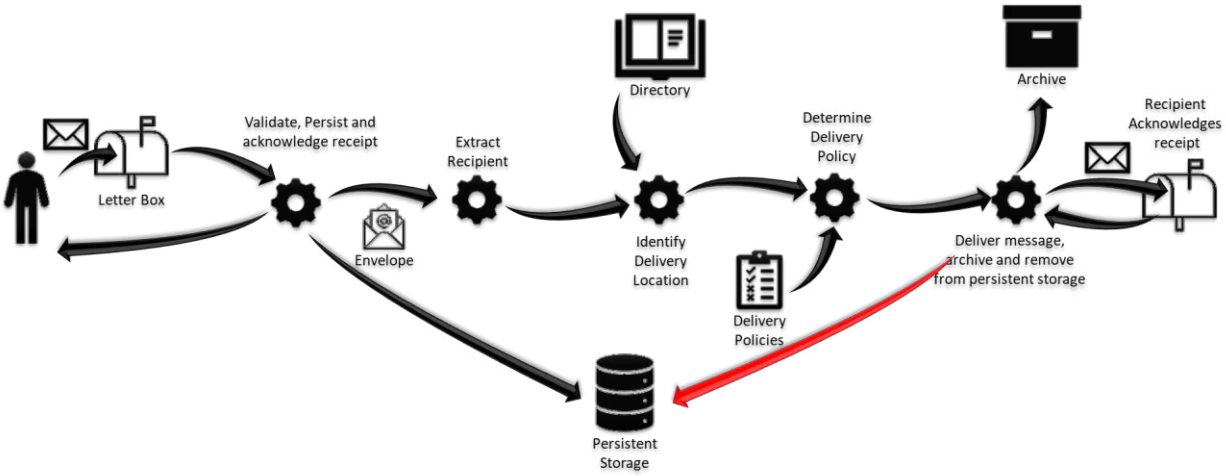

Figure 6 – Post Office Core Process Design

Working from left to right.

The post office will expose a post box. This will be a REST-based web service supporting a JSON message as a payload. Access to the service will be restricted to authorised users of the post office only.

As part of the synchronous process of receiving the message, the post office will validate the message is well formed and contains a valid envelope and that the originator and recipient are valid in the directory. Also, the originating network location of the message is consistent with the listed originator on the envelope to prevent spoofing. The message will then be persisted before acknowledging to the originator through a success HTTP response 200 confirming that delivery responsibility for that message has been accepted by the post office.

The recipient information is now taken from the envelope along with the message type, and the directory is interrogated to determine where the message should be delivered to.

For the message type, a delivery policy is determined by identifying how long to retry, in the event of failures for example, and what to do in the event a message cannot be delivered.

Finally, the post office connects to the recipient's letterbox and delivers the message. This is a synchronous REST service again with the message as the JSON payload. Upon successful response, HTTP 200, is received from the recipient, the post office can consider the message is delivered and the message, as well as its delivery status, are published to the archive. Finally, the message is removed from the persistence store.

This is a happy path process, and errors can occur at multiple points in the process of delivery from the initial receipt of the message to the final delivery. Should an error occur, the post office will either attempt to resolve the issue and continue with delivery or will respond to the originator with an error message.

Some key elements of the above workflow require further explanation as to the purpose and function below.

6.1 Persistence of Messages

The post office is a service providing a guaranteed delivery capability on behalf of the originator of a message. Should anything happen after taking receipt of a message and acknowledging it to the originator, such as a systems crash, that message must continue to be delivered after recovery. Many mechanisms exist to support this level of resilience, from full database persistence to simpler journalling systems, but the design should, in the event of a full system failure and switch to a standby solution, recover any unprocessed messages and continue with the processes of delivering them to the recipient.

For a specific letterbox, the messages must be delivered by the post office in the order they were received to ensure the integrity of the message processes, and any newly received messages should be added to the end of the list to be delivered. Consider a switch confirm and a switch completion is sent to a losing provider but processed in the wrong order, the completion would get rejected as the switch had not been confirmed yet, The originator would have to resend the completion message again.

In the event of continued failure to deliver the post office must apply a backoff mechanism. If deliveries to a specified recipient continue, then the post office should suspend the receipt of new messages for that recipient until communication is restored. This prevents the post office from being overloaded with unsent data and reduces the processing load. New transactions can then be received once connectivity has been restored and the backlog has been sufficiently cleared.

6.2 Directory

The directory has two functions. Firstly, it will provide the industry with a centralised location to store identifiers for industry bodies along with contact information, services supported and other information that will help consumers of the directory to interact with each other either through the TOTSCo hub or through other traditional means of communications.

The second function the directory performs is to hold the technical information regarding the messages the recipient supports, and the information required to deliver those messages to them, including any failover addresses in the event delivery cannot be made.

The directory is defined in more detail in a later section of this document.

6.3 Delivery Policies

Every message type will have a delivery policy, or if none is defined a default policy will be used. The purpose of the policy is to determine what should happen in the event of a failure to deliver a message.

For example, if a message has an associated SLA requiring delivery in 30 seconds, a policy could keep retrying sending the message every second for 30 seconds, then give up and send a failure to deliver the message back to the original sender.

Alternatively, the policy could state that delivery is required within 30 seconds, but to keep trying for up to 5 minutes. In that situation if the message fails to deliver in 30 seconds, a notification is sent to the sender of a delay in processing the message, and only after continued retries up to 5 minutes would a failure to deliver the message be sent to the sender.

Confirmation of delivery could also be provided on the delivery policy to inform the originator that a message has been delivered successfully.

A stricter policy may specify message delivery is mandatory, and the post office should continue to try delivering the message indefinitely. Periodic delay notifications should still be sent back to the message originator in this situation, reducing in frequency if the outage persists.

If a provider has to change a letterbox or create a new one because of a fault in their systems, once the configuration has been provided to the directory, the retry mechanisms should automatically try all letter boxes again for that message format, failures should not keep trying the originally selected letterbox if a failover policy has been configured with a group of letterboxes.

Should a recipient cease providing service and messages, the TOTSCo administrator must deregister that provider's ID from active message participation, and suspend the attempted delivery of their messages. Suitable processes and supporting tools would be required to remove any messages if a delivery location is no longer available to prevent messages from consuming resources in the system unnecessarily. Any removal would generate a failure to deliver notification back to the message originator unless they too were no longer a valid delivery location in which case the failure will simply be recorded for the audit store.

In the event of a letterbox needing to be replaced, a new letterbox must be created first and grouped with the original before then removing the existing one. Messages associated with a letterbox group will then automatically be redirected to the new letterbox to avoid lost transactions and causing unnecessary failure messages to the originator.

6.4 Message Routing

From originator to recipient, multiple delivery routes are required to be supported. For example, each message originator could generate messages from multiple systems, and likewise, each recipient may provide multiple letterboxes to deliver the messages. Some letterboxes may be specific to some message types only depending on the systems or process being followed. Within the post office, in the directory, all message formats, letterbox groups and letter boxes will be defined and associated so it is clear for any messages sent how they should be routed and delivered to another provider.

The following diagram lists all of the elements affecting the routing of the message with an example configuration.

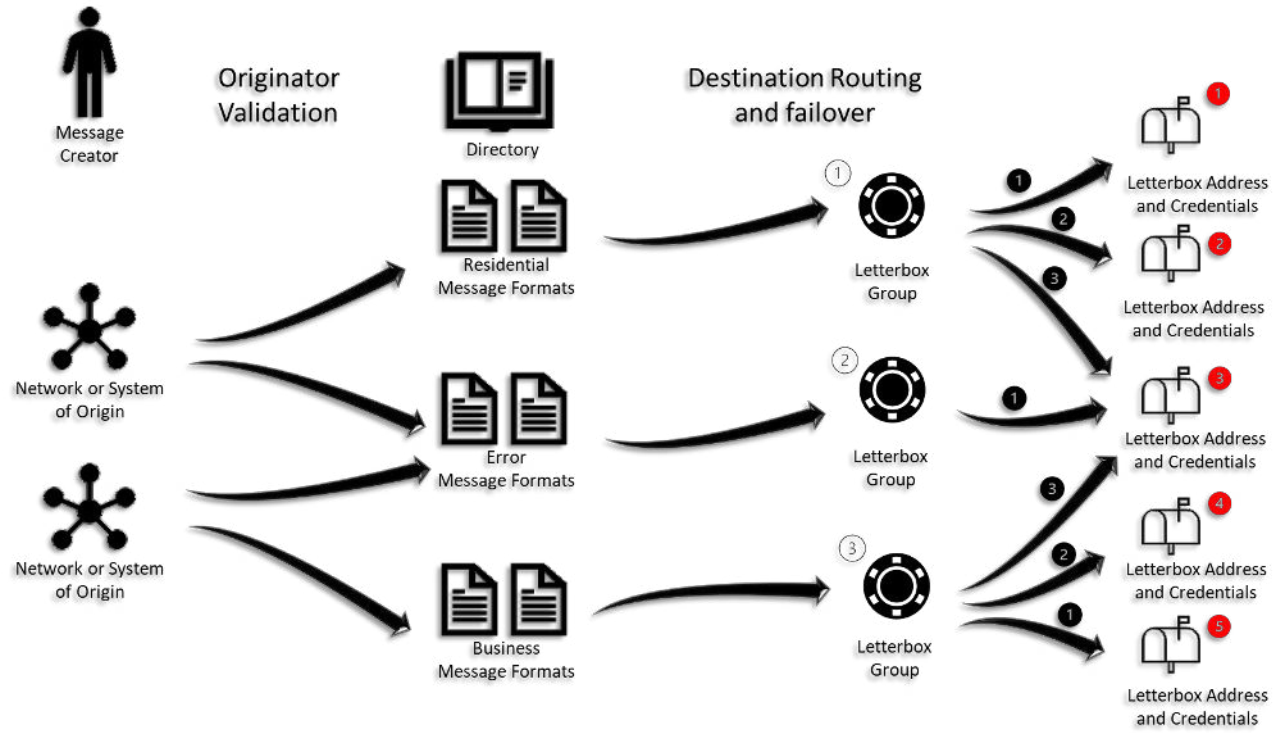

Figure 7 – Example Message Routing Rules

Taking the model above, the directory will contain a list of letter boxes. These represent the web service addresses and the credentials needed to communicate with them.

A letterbox group is simply a collection of letterboxes supporting the same service capability. So for example, there may be a group for consumer message processing and another group for business. Each group may point to multiple letter boxes for resilience.

Details of the letterbox group and letterbox specifications can be found in section 8, Directory.

So from the figure above, residential switching messages are configured to route to the letterbox group ① which can deliver those messages to any one of three letterboxes, ❶❷❸, developed to support those messages in order. Those letterboxes are used in the specified order, ❶❷❸, so if connectivity to letterbox ❶ is lost then it will try sending to ❷ and so on.

Business messages are configured to deliver to letterbox group ③ and that delivers to letterboxes ❺❹❸, again in that order defined by the list order.

Finally, the letterbox group ② is configured to only send to letterbox ❸. This may be an offline or error-handling letterbox used when all other letterboxes are unavailable.

This is probably an over-elaborate scenario, most businesses are likely to only associate a single letter box with a letterbox group, but the configuration allows for a failover mechanism if required.

Each retailer may have multiple originator systems generating requests to the post office, and there must be a security definition for that access within the post office to determine for each originator which source IP addresses are allowed access from that location. There is a requirement

that all requests be identifiable by their originator in a way that can be reliably validated. For example, IP addresses, DNS names, credentials etc. This is to enable a secure connection policy to be defined.

6.4.1 Failure Message Routing

Every message should translate to a letterbox group specified by the recipient ID where those messages will be processed. However, in the event of the post office either being unable to identify a letterbox group, not having any letterboxes or the letterbox not responding, the post office will create a failure message back to the originator.

Note, that this is not the asynchronous validation of the message which concentrates on identifying faults with the message structure and envelope, but after acceptance with the onward routing and delivery of the message.

In this situation, the delivery failure message must be sent to the originator, but because this is not a response to a specific message format it would not have dedicated routing. Therefore, the directory will provide the means to specify where delivery failure messages should be returned to the message originator.

The originator has the option to configure how this is managed. Either by designating a letterbox group for all failures, a group for failures originating from a specified originating network location (DNS/IP etc.), or a group for errors relating to the message format that was trying to be delivered.

The configuration rules will apply in priority order as follows.

1. See if there is a letterbox group configured for failures against a specific original message format.
2. See if there is a letterbox group configured against the originating point (DNS/IP) of the message.
3. Default to a common letterbox group to deliver failure messages to.

As a message is received and processed through the post office, it should be tagged with the origination location information so that should a failure occur later in processing the return route for the reply can be properly defined.

It is the responsibility of the organisational entity that is defining all of its origination points and recipient points to configure appropriately so that it can send and receive messages, including error routing errors, and process them effectively and reliably.

6.5 Post Office Management and Monitoring

The post office will be required to provide tools to support the management and processing of messages.

Functions would include, but not be limited to the ability to place a recipient of messages on suspension so no further messages can be received for them. Being able to manually reject pending transactions as undeliverable, with an appropriate error indicating these were rejected by the operator of the post office.

Functionality can be granular, for example marking a single letterbox or an entire letterbox group as unavailable.

These tools should be secured by role and available in the post office admin portal.

7. Security Principals of all TOTSCo Hub components

For the post office application, portals and all associated service and data stores, the following set of core principles will be applied as a minimum.

- Data at rest and in motion encryption
  - To protect any information from data breaches in line with GDPR
- Intrusion detection
  - For identifying targeted attacks, weaknesses etc.
- Event logging
  - To record all events.
- Firewall protection
  - Perimeter firewall for stateful inspection
  - Internal firewalls for application and data protection
- Client and Server authentication
  - Are the client and server known to each other
- Service Authorisation
  - Is the client or server entitled to execute the service being invoked
- DOS Protection
  - To prevent disabling the service through flooding the post office

7.1 API Security

API calls to the TOTSCo Hub will be required to comply with a set of agreed security standards and principles, these are to be defined.

For initial guidance and discussion, the following is a list of security areas and technologies to be considered.

- Open Standards
- Client and Server authentication

It is expected that the most likely mechanism for authentication will be OAuth 2.0, but others may also be made available for consideration.

All API calls would be expected to be RESTful, and utilise JSON as a document body. URL encoding of API parameters may be used where appropriate but should not be used for message contents where onward processing of the message is required, the message body alone must convey all required information for a transaction.

TOTSCO

8. Directory

The directory is a master central data store containing information on all parties participating in communications via the post office, processes that utilise the post office, or other services in future as well.

For a one-touch switch, this involves defining a list RCPIDs, but as previously stated the directory should be capable of holding lists of multiple ID types and supporting information for the services that type will utilise. Therefore, the top-level element of the directory is the list type.

Immediate additional uses identified for the directory would be to provide a central source for managing not only RCPIDs but RIDs and CUPIDs to name only two. Though OFCOM maintains the master list of the identities, it does not provide any further useful information that parties utilising that information can make use of, for example, telephone support numbers.

Within each list type will be the IDs themselves, and objects representing the information held. The principal object associated with the ID is the definition of that ID, the public name representing the organisation that ID represents. Associated with that root identity object will be a list of trading names so the entity can be identified if they are representing multiple brands for example.

Where information is duplicated across lists, for example, contact details, organisation names etc., that information should only be stored once and linked to internally so to prevent the chances of data becoming out of sync.

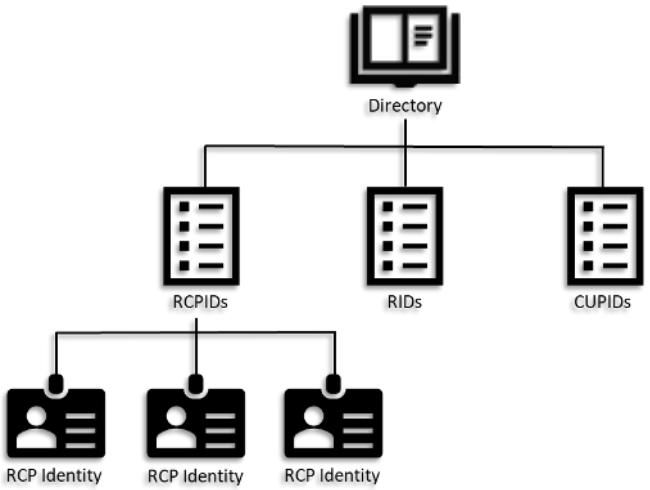

Figure 8 – Directory top level

For all lists, the key first object will be contact information. This should be a simple list providing named roles or business functions, and one or more contact details such as phone number or email address. Information in this object should be considered restricted for the use of all consumers of the directory to view and use in support of the business processes the directory is providing support for. Also, this information should not be duplicated across lists, rather a single

master copy of the company and contact information should be maintained for that organisation and referenced in each list so that the details only have to be updated once to update all lists.

It is the responsibility of the retailer to ensure all of their information is current and accurate.

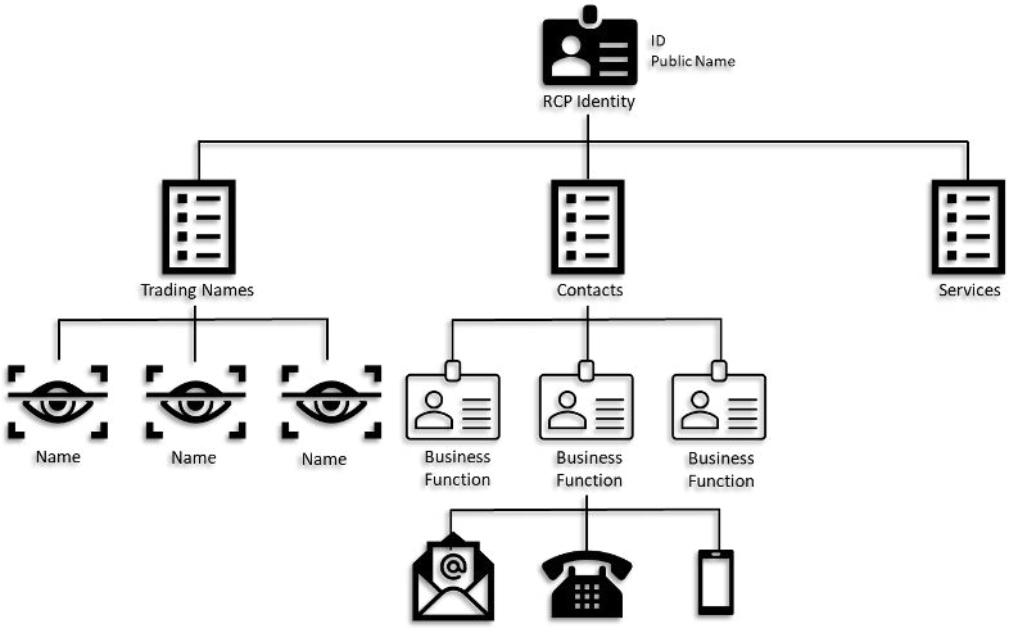

Figure 9 – Directory of Common Structures

Alongside the company information are the services that the entity supports and any configuration information necessary to support the use of that service. Services will be much more specific to the service being provided and each one may have entirely different substructures in each list. The directory must support this hierarchical organisation of information.

The services will be a list of known supported services by the directory owner as each service will have its requirements and security policies around the access to the information within those structures.

For example, for CUPIDs, a list of supported porting transactions could be defined, such as SPX which will be used in switching. This would remove the need for separate dedicated lists for these functions and keep everything contained within the single directory model.

To support switching, a switching service will be specified under the RCPID in the following structure.

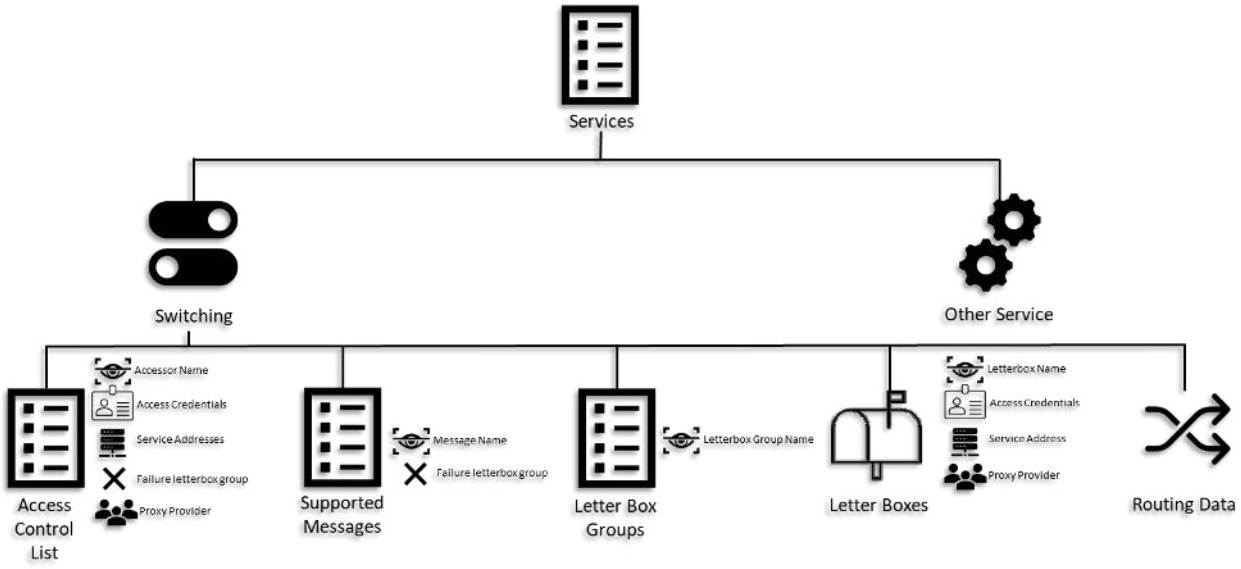

Figure 10 – Directory Service Structure

For messages sent to the post office's letter box, an Access Control List must be defined for the RCPID which specifies the identity and any associated security-related information necessary to verify the originator of the message and permit its processing. If an entity uses a proxy, [redacted] [Redacted] that can be defined here instead of a direct service address.

The supported messages list specifies the names of all switching message formats supported by this RCPID. For example, an RCP may only support consumer switching and not business or vice versa.

Letterbox groups provide a mechanism to group letterboxes together for failover purposes and to designate routes for specific message formats.

Letterboxes define the delivery address for a specific IP connection on a recipient's network where messages will be sent. The configuration of this element will define the access credentials, public keys etc. to use the letterbox API. Again, if an entity uses a proxy [redacted] [Redacted] that can be defined here instead of a direct service address.

A single letterbox group can be specified as a default in case no routing has been configured for a specific message format. However, this can be omitted if the rejection of unrecognised message formats at the post office level would be preferred.

The final entity in the above diagram is the routing data, shown below, this specifies what messages can be sent or received from where.

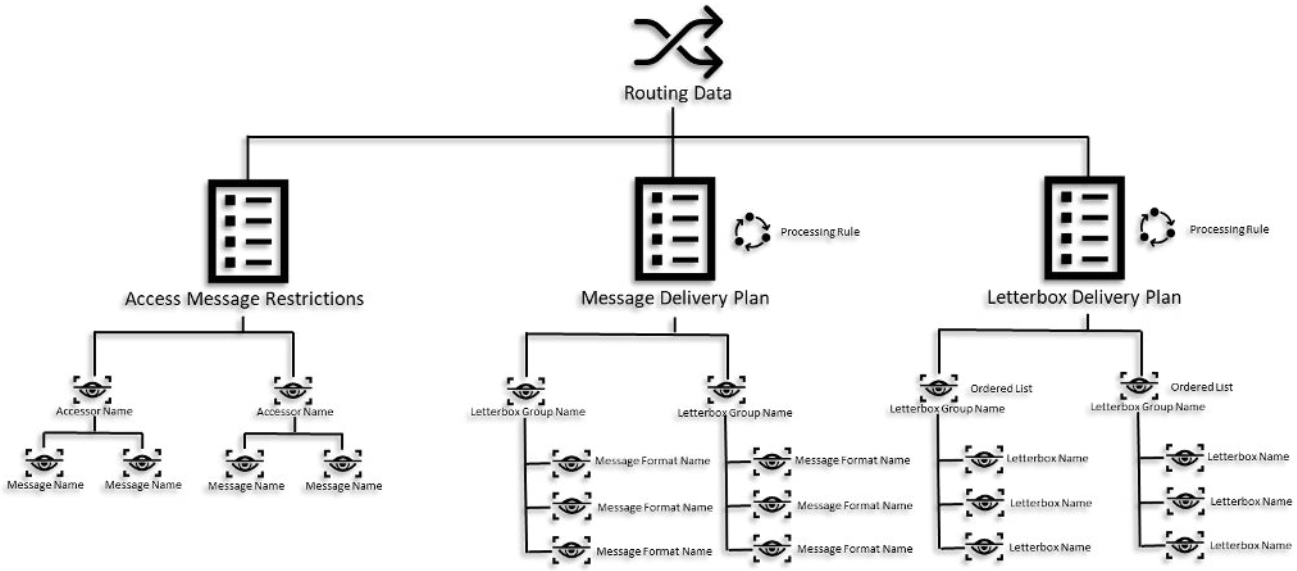

Figure 11 – Directory Routing Data

So again, working left to right above, the access message restriction rules list the messages that should be allowed from each accessor in the ACL. This is a security policy implementation to prevent unauthorised parties from sending messages that are not expected from that location.

Next is the message delivery plan. This defines the list of letterbox groups and the messages that should be routed to them.

Finally, the letterbox plan specifies the list of letterboxes for each letterbox group and the order in which delivery should be attempted.

The design of the directory structure, and its storage, should consider the need to provide services to parties associated with the directory to download the exportable elements of the directory. The mechanism for export would be in JSON format which would allow new elements to be added to the directory structure in future without breaking the existing processing of the documents. JSON parsers have very good options for ignoring elements of data the recipient is not expecting, even in an environment where mapping messages onto object structures, hence this flexibility to expand message content without breaking existing processing logic will be of great benefit to the industry when rolling out new capabilities.

In storage, and processing the directory contents must be indexed for performant searching by the relative components of the post office, or any other system consuming data from the directory.

It would be expected that the directory will have an associated schema that would define the list structure, and the nature and security policy associated with each element. For example, identifying that in the RCPID branch of the directory, the list was public except for elements under the switching node where all of a user's confidential information is held. This would be applied when users of the directory are obtaining a copy of it for their service caches.

9. Other Configuration Information

Supporting service within the TOTSCo Hub, various services will require configuration that is applied by TOTSCo and not defined by the users of the services. These configurations should be stored separately from the directory for use by the services that consume them.

9.1 Post Office Message Delivery Policies

For the post office service, a message delivery policy will be configured for every message type, defined and agreed upon by TOTSCo in conjunction with its members.

The policies define what should happen to messages that cannot be delivered, whether to retry them, how often to retry per letterbox or letterbox group, how long it should retry, and how it should notify the originator of the message if it was not delivered, whether to give up after some time or several retries or whether the message must continue to be retried until it can be delivered.

Note, that a message rejected by a letterbox synchronously may be considered the same as a failed delivery, subject to the error information returned. For example, a letterbox may remain active and in service but is unable to currently process messages while maintenance is being performed. In such a case the post office may rightly retry sending, failing over to another letterbox. However, if the letterbox rejects the message because it thinks the message is malformed, that should result in a failure to deliver message back to the originator with the reason the recipient rejected the message content.

10. Archiving Transactions

As messages are processed through the post office, on delivery of that message it will be archived, along with information associated with that transaction.

The message should be delivered to an archival ingestion process which needs to identify extra information from the message to store in the database so that it can be easily searched for if needed.

Therefore, an archival service is required, and message handlers for each message type.

An example of the specific information needed from the message to be searchable in the archive is the originator and destination list type and ID, any correlation IDs present, and the SOR from the message body.

There may be a further need for other attributes to be extracted to facilitate faster searching and reporting, all of which must support plain text searching.

The archive then supports the reporting and auditing processes of this architecture.

11. TOTSCo Hub Admin Portal

To support all of the functions and services of the TOTSCo Hub solution, an admin portal will be required to manage all configuration settings. The portal will provide a single point for all

configuration of the directory and policy information used by the Post Office and any other supporting services.

Broad browser support will be required for the circa 7000+ organisations that will use the portal to administer their service configurations, principally Microsoft Edge, Google Chrome and Firefox.

All changes made within the admin portal will be audited and logged for security purposes, and all access to the portal will be secured.

11.1 End User Admin Functions

As end users of the TOTSCo Hub, those users must create the configuration of their infrastructure, what they support and how they want data handled within the various TOTSCo Hub components. An end-user may have access to manage information for multiple IDs on multiple lists if, for example, they are an agent or wholesaler providing services on behalf of retailers.

The directory defines the list of elements the end user admin functions need to manage, other than the list and list ID creation activities which will be managed by TOTSCo as those IDs are applied for and allocated.

11.2 TOTSCo Admin Functions

As the operator of the TOTSCo Hub, all core operations for managing settings not related to a specific service consumer will be defined through the TOTSCo Admin Functions.

This list includes but is not restricted to the following information.

- Supported Message Formats
- Post Office Message Delivery Policies
- Association of Delivery Policies to Message Formats
- Creation and management of lists
    - RCPID List
    - RID List
    - CUPID List
- List Root Identities and Security Information
    - Creating RCPIDs for retailers
    - Loading OFCOM RID and CUPID Lists
- End user management
    - Creating initial security credentials for end users to log in and manage their data

TOTSCO

- Associating end users to RCPIDs, RIDs, CUPIDs etc.
- Being able to reset security information

12. [redacted] [Redacted]

[redacted]

12.1 [redacted] [Redacted]

[redacted]

[redacted]

[redacted]

[redacted]

[redacted]

12.2 [redacted] [Redacted]

[redacted]

[redacted]

[Redacted]

12.2.1 [Redacted]

[Redacted]

12.2.2 [Redacted]

[Redacted]

12.2.3 [Redacted]

Redacted]

TOTSCO

12.2.4 [Redacted] [Redacted]

[Redacted]

12.2.5 [Redacted] [Redacted]

[Redacted]

12.2.6 [Redacted] [Redacted]

[Redacted]

12.3 [Redacted] [Redacted]

TOTSCO

[redacted]

[Redacted]

12.4 [redacted] [Redacted]

[redacted] [Redacted]

13. [redacted] [Redacted]

[redacted] [Redacted]

14. TOTSCo Hub Operational Dashboard

To support operational management and visibility of the activities of the TOTSCo hub, and all of its services and functions, a dashboard will be provided that represents the current status of all services, including recent usage levels, transaction processing throughputs etc. for monitoring.

The dashboard should at least be available to TOTSCo operational staff, but ideally also available to query by hub users as well to show data for their activities.

Examples of information that should be available on the dashboard would include numbers of active transactions in the hub, uptime reports by hub component and the last known status of each attached RCPID.

TOTSCo users would require more detailed system information, for example, CPU utilisation, bandwidth usage, database size etc. depending on the deployment of the hub

The dashboard should also have the means to display any outage information, along with any ETA for resolution if a service is down for any reason.

TOTSCO

15. Switching Data Management

The processes around switching will generate significant amounts of data that must be retained for guaranteed delivery and auditing.

All data stored for any function of the TOTSCo hub must be done per the security principles defined in section 6.5, Post Office Management and Monitoring.

15.1 Data Retention

Retention and archival policies must be definable for all transactional and auditable data generated or processed by any element of the TOTSCo hub.

Retention periods should be configurable in days.

Once data has reached its retention age, it should be archived or purged per the retention policy for that data set.

15.2 Data Archival

Data archival refers to the storage of data outside of the TOTSCo hub where that data has exceeded its retention period and is no longer needed within the TOTSCo hub for operational purposes but is required to be retained for operational support, regulatory or legal reasons.

Archiving data provides a means to offline data to reduce processing overheads of the core TOTSCo hub systems, while still providing access to it for regulatory and reporting purposes.

Data will have a storage period in archival separate from the retention periods of live data. Once that storage period is exceeded the data will be purged.

15.3 Data Purging

Purge processes should remove all references to a data element, ensuring any referential integrity is not compromised.

Purging will be applied as per the data retention policies for each data set stored within the TOTSCo hub.

Audit records of purges should be held recording the data set removed, how many records, date and time and the date range from that set.

15.4 Event Logging and Auditing

All actions performed by the post office, or one of the associated portals or other components of the TOTSCo hub must be logged and an audit capability provided to be able to trace those events, for viewing or reporting as necessary by authorised users.

All events must be sequenced, timestamped, and stored securely to prevent tampering or unauthorised access.

All logging from all TOTSCo hub functions must be centralised and capable of free text searching.

16. Reporting and Regulatory

To support the operational and support requirements of the TOTSCo hub, a portal will be provided with the ability to create reports using the data stored in the TOTSCo hub.

The data available will be restricted by the user requesting the reports, with retailers only being able to report on their switch cases, either as the gaining or losing provider. TOTSCo operators or regulatory representatives will be permitted to create reports on all switching cases.

Reports should be capable of being pre-defined and generated automatically on a schedule or demand.

Reports should be provided in a PDF format, and if appropriate, a CSV format for an extracted data set, compressed and encrypted. Reports can be created and stored locally for the intended recipient or automatically emailed to a designated address.

16.1 TOTSCo Management Reporting

Examples of management reports will be provided later once requirements are formulated but would represent both summary and detailed reporting. Examples could be as follows.

- Number of switch cases and states by all, or selected RCPIDs over a selected period.
- List of active switch cases for an RCPID
- Number of cancelled switch cases by RCPID
- Matching response times by RCPID by month
- Detailed switch case report for a specified SOR, sequence of events.

16.2 TOTSCo Operational Reporting

Operational reporting refers to the reporting of log-related information rather than switch-related activities.

These reports also should cover system status reports, including uptime, availability, and transaction response times for example.

All systems in the TOTSCo hub must produce or expose the means to capture the operational data required to meet the needs of the reporting requirements, as well as the operational dashboard.

The extent of what information would be required in operational reports is to be defined.

TOTSCO

16.3 End User Reporting

Very similar to the TOTSCo management reporting, but restricted to only allow the viewing of information associated to switch cases associated with the logged-in users' RCPID.

A list of required reports will be defined during the detailed design.

17. Transaction Accounting

TOTSCo's operating model will define a financial model for users of the TOTSCo hub to charge for its use. The exact model is still to be defined but could be affected by several factors, for example, the number of switch cases raised as a gaining provider, the total number of messages sent to the post office, and the number of customers the hub is servicing on behalf of the customer.

Whatever model is defined, the transaction accounting solution must be capable of recording all instances of that chargeable element during post office processing and forwarding it for data collection in the transaction accounting system.

Once there, the system will aggregate the chargeable events, applying any discounting rules if any are defined and then documentation to support an invoice on behalf of TOTSCo for sending to the retailers.

The accounting solution should keep a track of those documents sent to the retailers and allow recording of status, outstanding, overdue, paid etc. to support the accounting system.

This is not a requirement for a full finance management system, only a system to capture and aggregate chargeable events to facilitate the raising of invoices into an accounting system.

The exact events from the post office have not yet been defined, but could be linked with the archival process.

18. [redacted] [Redacted]

[redacted]

TOTSCO

[redacted] [Redacted]

19. Service Monitoring

A real-time service monitoring capability is required, informing the dashboard on systems availability within the TOTSCo hub, and being capable of generating alerts and notifications/email events to TOTSCo hub operators in the event of service outages.

20. Recipient Letterbox API

Parties utilising the services of the post office, must not only provide the means to deliver messages to the post office's letterbox but must also provide their letterbox for messages to be sent back to themselves. A party may provide multiple letterboxes for failover or for handling specific message types as necessary.

The letterbox API will conform to the same specifications as the post offices letterbox, HTTP with REST as the message body, but the access credentials may differ depending on the security scheme implemented by individual originators, OAUTH 2.0 would be preferable, but several industry agreed specifications will be supported for multiple web access security standards.

The definition of how to access that letterbox will be defined in the service directory along with any public encryption keys, API access keys or certificates as necessary for the security mechanism selected. This information will only be accessible to the post office itself, and not shared with any other party.

A retailer will be solely responsible for maintaining the configuration information related to their letterboxes, service origination points etc. in the post office admin portal.

The API must synchronously validate the message structure and envelope only, and once the API acknowledges receipt back to the hub, only then can the message be processed fully.

The hub will already have confirmed a correctly valid structure, so if the recipient should determine that the body of the JSON is in a format it cannot understand, then it should generate an asynchronous error message back to the originator including their original correlation ID and the fault information.

The synchronous response time for the API must be as quick as possible to free up hub resources and allow as high a throughput of message delivery as possible, so keeping message processing out of the letterbox API is a key design requirement.

Once a message is accepted by the API and acknowledged, it is entirely up to the retail CP to define the technical behaviour of the service within their core systems, whether it onward routes the message to other systems, or processes the messages as they arrive, the specification will not dictate individual implementation rules.

TOTSCO

21. API Interfaces

The following section lists most of the expected APIs to be provided by the TOTSCo hub in support of direct connection from Retailers [redacted]

API Security standards will be applied per the details in section 6.5, Post Office Management and Monitoring.

The API URL format for the APIs provided by the TOTSCo hub will conform to the following convention.

https://postoffice.targethostdomain/apiname/method

The target host will be determined by the hub vendor and TOTSCo, depending on the domain name used. The API names will be defined once all APIs have been formalised as will the method.

All functional APIs will use POST as the method of request. Any APIs required to support GET will only apply to static data and support caching of results, none are anticipated in the current design.

This is not expected to be a complete list, but identifies the core capabilities expected.

21.1 Directory List API

The directory listing API will allow users to obtain the contents of a list from the directory. The API will allow the specified list to be provided, and by return, the API will return a JSON document representing the directory structure for all identities within the specified list contents but only including elements that are allowed to be exported, so no system, security or routing information.

This API supports the use case in the OTS process document for retailers to obtain an updated list of RCPIDs, their names etc. so that their sales flows, portals and business systems have access to the full list of RCPs for customers to select from and search when deciding to switch. By allowing the individual RCP to cache this information, it takes a significant load off of the TOTSCo hub.

In addition, being able to cache this information locally provides the information the providers will need in the event direct contact with a provider is required, more likely during business switching. This information can be loaded into their core business systems as contacts for example as necessary.

Every retailer providing switching services must be registered in the RCP Directory and be provided with an RCPID.

It is expected that most providers will call the API daily to obtain the latest list of information and to update their internal data caches.

The URL for the letterbox API will be as follows.

https://postoffice.targethostdomain/directory/read

TOTSCO

21.2 Letterbox API

The TOTSCo hub's core purpose is to deliver messages, and for this, it will expose a single API.

The letterbox API will be identical in specification to the recipient letterbox API and will receive a JSON payload of the message to be delivered by the post office to the addressed recipient. The API itself is very simple and only requires the JSON message to be provided within the REST API body.

There may be some additional information required to be added to the URL for the API to process the message, for example, any authentication information. That information will only be for use by the letterbox API itself and will not be passed on to the recipient of the message or used in its internal processing in the post office.

The JSON body of the message is made up of two parts, the envelope that the letterbox API reads and processes, and the Message content itself which the API will ignore and not validate at all other than structurally to ensure compliance with JSON document standards.

The following is a skeleton JSON message format supported and processed by the letter box API.

```
{
    "envelope": {
        "source": {
            "type": "RCPID",
            "identity": "ABCD",
            "correlationID": "XYZ987"
        },
        "destination": {
            "type": "RCPID",
            "identity": "ABCD",
            "correlationID": "ABC123",
        },
    },
    "messageFormatName": {
        ….
    }
}
```

The envelope structure contains only the basic information needed to deliver the message. It identifies the originator of the message which the service will validate against the senders' credentials to ensure the messages are not being spoofed, the destination for the message and the message format name. The name of the message format is validated against those messages allowed to be sent by the sender RCPID to add another layer of security.

The message format name will be different for every type of transaction passing through the post office and will do multiple things. For the letterbox API, it will identify how the message will be routed to the recipient. This will include which letterbox to send to as well as the delivery policy for the message, identifying the timeout period for delivery, retry options, and actions to perform on a failure to deliver etc., these will all be defined within the admin portal for every message Format.

The source and destination type and identity both refer to entities defined in the directory and using this nomenclature allows for the delivery and processing of messages for many different industry

functions, not just switching for RCPIDs. So for example a porting transaction could be created with “CUPID” as the type and a CUPID value as the identity for both source and destination.

The correlation ID must always be provided in the source information of every message sent (except for failure messages from the post office) and must always be returned by the recipient in the destination information. A recipient responding to a message must also include a new correlation ID in the source information as well. A new message would never contain a correlation ID in the destination element of the JSON.

The reason every message must have a correlation ID is that, in the event of a failure to deliver a message, the post office can return a delivery failure notification specifying the originator's correlation ID so that it knows how to handle that failure. Think of this much like a consignment number in postal terms, it allows you to understand which parcel you are talking about when information about it is sent back to you.

The letterbox API is a synchronous receiver API only. In other words, its role is to accept the message reliably and securely, and to respond to the sender acknowledging receipt of it. The API will validate the message and any errors with the envelope, JSON structure, and unsupported message formats will all be responded to synchronously. However, once validated the letterbox API will respond to the sender confirming receipt synchronously and only then will it pass it either to the post office routing functions for onward delivery of the message or in the case of a consumer of the post office it will process the message in within their business systems. If a response to the message is required, this will happen asynchronously either with the recipient of the message responding, or the post office generating automated replies if there is a failure to deliver for example.

JSON element	Description	Format
envelope	A container defining the delivery information for any associated message.	Object
source	A container defines the originator of the message and represents the return address for any message requiring a response.	Object
destination	A container representing the destination of the message and used by the Post Office to identify the correct letter box to deliver it to.	Object
type	The name of the directory list where the identities can be found and validated.	String
Identity	The identity of the sending or receiving entity for the message as defined in the directory list selected.	String
correlationID	A string of characters supplied by the message originator, either of their creation as the originator of a message, or from a message sent from another party to allow them to identify the response concerning the request previously sent.	String

The messageFormatName container will differ based on the messaging process being followed and these formats are described in the documentation associated with those processes.

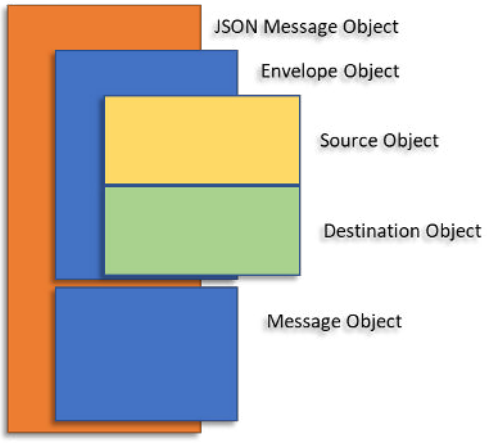

Figure 12 – Post Office JSON Message Structure

The container structure of a Post Office message is displayed above, the message object is separated from the envelope to allow changes in the content of either structure without affecting each other.

The URL for the letterbox API will be as follows.

https://postoffice.targethostdomain/letterbox/post

22. Message Formats

The industry hub API has defined the envelope of the JSON messages sent via the hub. These are the only parts of the JSON message the hub will be responsible for understanding. Within each envelope, there will be a messageFormat that will define the structure of the message body, and this is the element that the recipient of the message must understand. Both parts together form the entire JSON message.

There may be hundreds of supported message formats in the future if the use of the hub is extended to other cross-industry processes, for example, number porting.

The messaging specifications for One Touch Switch and GPL switching are available in separate documents as these will evolve and be controlled separately from the TOTSCo hub itself which is intended to provide a largely generic messaging solution with some specific components to support switching processes. Some elements of the TOTSCo hub will be required to understand message formats, for example, the archive and invoicing solutions that will need to identify messages that require charging.

The message formats described below are those explicitly created and sent by the Post Office.

22.1 Post Office Faults and Messages

In the event of the post office being unable to deliver a message to its intended recipient, or if status update messages are sent, it will create a message to the originator of the original message by providing the following information.

```
{
    " envelope ": {
        …
    },
    "postOfficeMessage": {
        "code": "9001",
        "text": " Unable to deliver the message to the destination, no valid route.",
        "severity": "failure"
    }
}
```

The error code describes the fault reason and the correlation ID of the originating message will be provided to allow the sender of the message to identify what failed and the actions that they wish to take to resolve the issue.

JSON element	Description	Format
postOfficeMessage	Container for messages from the post office	Object
code	A number that represents the nature of the fault and can be used by the message originator to determine remedial action.	Integer
text	A description of the associated response code	String
severity	An indicator of the nature of the message about the processing of the originators' message.	String

22.1 Response Codes

The following table defines the list of response codes the post office will generate in the event of an error processing a message.

Code	Text	Severity
9000	**Unknown or missing destination Type**	**Failure**
9001	**Unknown or missing destination ID**	**Failure**
9002	**Unknown or invalid source Type**	**Failure**
9003	**Unknown or invalid source ID**	**Failure**
9004	**Source type and ID not permitted from originating location**	**Failure**
9005	Unable to deliver the message to the destination, no valid route.	Failure
9006	Unable to deliver the message to the destination, rejected, invalid message format	Failure
9007	Recipient rejected message	Failure
9008	Unable to deliver the message to the destination, timed out.	Failure
9009	The message has not been delivered to the destination but will be retried.	Info

23. Glossary

Term	Definition
CP	Communications Provider
GPL	Gaining Provider Led

IAS	Internet Access Service
NBICS	Number Based Interpersonal Communications Service
OTS	One Touch Switch
RCP	Retail Communication Provider
RCPID	Retail Communication Provider ID
SOR	Switch Order Reference (UUID)

End of Document